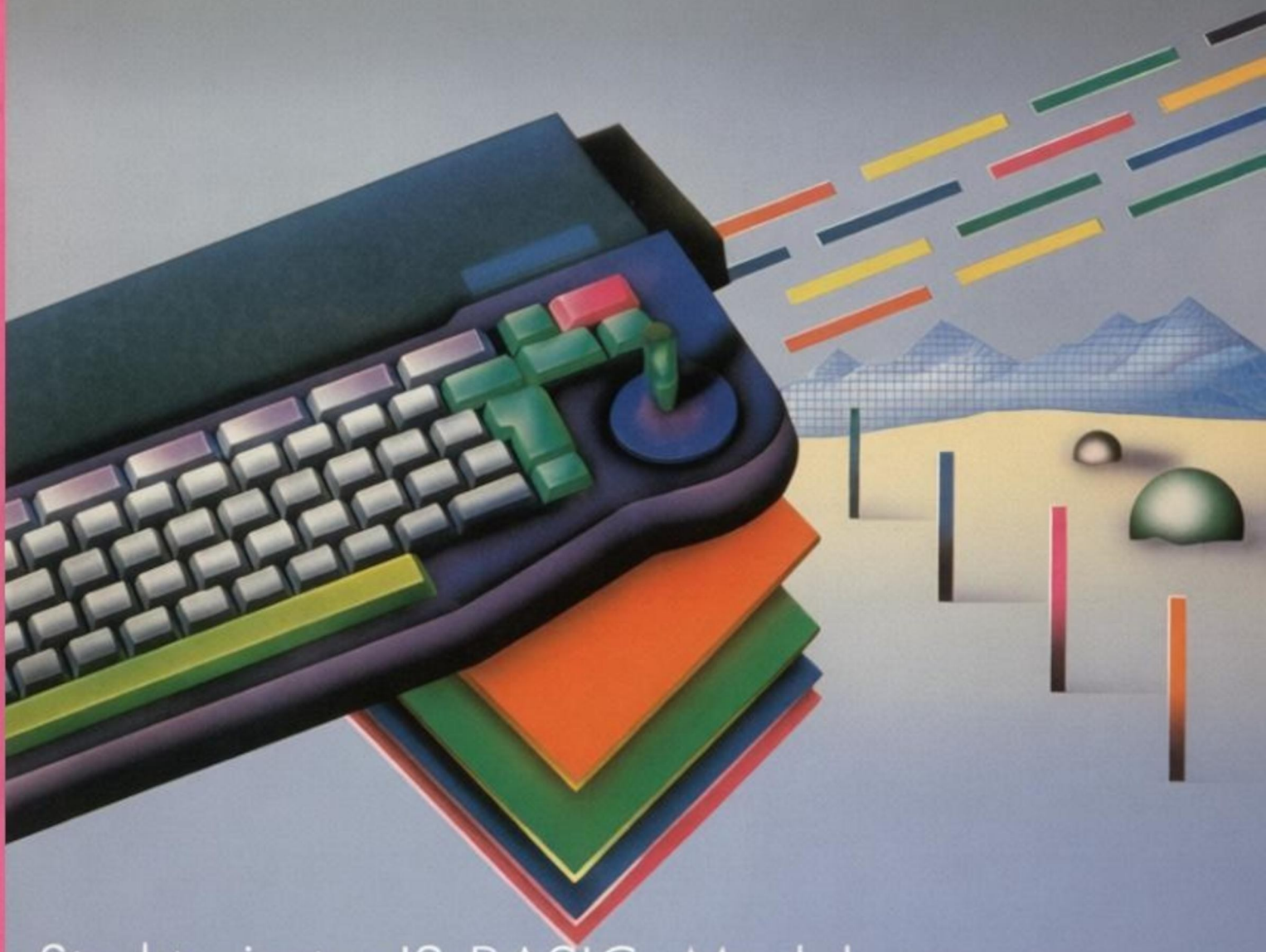
Computer-Programme: Enterprise, Ausgabe 1

CHIP

Computer-Programme Enterprise 64 K/128 K

18,– DM



Strukturiertes IS-BASIC · Modularer Progrommaufbau · 3D-Grafik · Kreatives Gestalten · Künstliche Intelligenz · Teachware Small Business · Stereo-Sound · Tricks

CHIP-SPECIAL

CHIP-SPECIAL

Telekommunikation

für Home- und Personal-Computer · Datenfernübertragung mit Akustikkoppler und Modem · Hacker-Erfahrung · Kosten · Zukunfts-perspektiven · Datex-P- und Mailbox-Nummern

Mit Programmen für DFÜ

24,— DM · 81890/85004

Eine Publikation von PE
Das populäre Magazin für Elektronik und Technik

Werkzeuge für die eigene Programm-Bibliothek

Mit der vorliegenden Ausgabe halten Sie eine Programm-Sammlung in Händen, die sich aufgrund Ihres modularen Aufbaus hervorragend dazu eignet, eine eigene Programm-Bibliothek aufzubauen.

Unsere Programmierer haben keine Mühe gescheut, um Ihnen, von alledem was der Enterprise kann, möglichst viel zu bieten. Schließlich ist der in die Tastatur eingebaute Joystick, mit dem der Cursor schnell und präzise positioniert werden kann, bestens geeignet zur Bearbeitung von Texten und Programmen.

Das in einem Steck-Modul untergebrachte IS-BASIC sorgt nicht nur dafür, daß alle Hinweise des Textsystems in Deutsch sind, sondern ermöglicht zudem strukturiertes Programmieren. Der Befehl LIST erkennt nämlich Strukturen in BASIC-Programmen und rückt einzelne Blöcke im Listing automatisch ein.

Von Vorteil bei diesem Rechner ist zudem, daß lange Listings und andere Ausgaben nach ihrem Verschwinden vom Bildschirm bequem wieder zurückgeholt werden können, indem man den Cursor mit dem Joystick nach oben bewegt - der Computer speichert nämlich jeweils mehrere Bildschirminhalte. Auf diese Weise lassen sich mehrere Programme gleichzeitig im Rechner unterbringen, jedes mit einer Nummer oder einem Namen versehen.

Die Sound-Programmierung stützt sich auf vier Tongeneratoren. Als Programmierer hat man daher viele Freiheiten in der Festlegung von Hüllkurven und Wellenformen. Mit den beiden Lautsprecherausgängen lassen sich ohne Schwierigkeiten Stereoklänge erzeugen. Ein Tonstapel ermöglicht es, ohne großen Aufwand Hintergrund-Musik ins Programm einzubauen, ohne das Programm selbst zu verändern.

Der Enterprise teilt die 128 Kilo-Bytes in Segmente zu je 16 KB auf. Das hat den Vorteil, daß man mehrere Programme gleichzeitig im Speicher halten kann, wobei für das erste Programm 48 KB und für jede weitere Programmebene 32 KB Speicherplatz zur Verfügung stehen. Sämtliche, in dieser Ausgabe veröffentlichten Programme sind mit der eigehefteten Antwortkarte am Heftende auf 3,5-Zoll-Diskette zu haben.

Viel Vergnügen im Umgang mit IF-THEN-ELSE, WHILE, SELECT und DO-LOOP wünscht Ihnen Ihre

Redaktion CHIP-SPECIAL

CHIP SPECIAL

Neue Medien

Bildschirmtext · Videotext · Kabelfernsehen
Mailbox · Pay-TV · Satelliten-Empfang
HiFi-Empfang · Bildplatte · Bildtelefon
Zellular-Telefon · Bigfon
Home-Roboter

24,– DM · 81890/85003

Eine Publikation von PE
Das populäre Magazin für Elektronik und Technik

Inhaltsverzeichnis

Grafik	6	Sierpinski-Kurven
	8	Hilbert-Kurven
Lernen	9	Vokabeltrainer
Mathematik	18	Integralrechnen
Hobby	21	Mein Biorhythmus
	24	Life is life
Small Business	29	Adressen
Datenbank	37	Universal-Datei
Statistik	48	Polynome
Steuererklärung	57	Abschreibung
Physik	63	Teilchenbahnen
Biologie	69	Räuber-Beute-Population
Rekursion	73	Hanoi-Türme in IS-BASIC
Denkspiel	74	Logic
Kartenspiel	80	Skatrunde
Adventure	87	Der Rote Eric
Strategie	102	Enterprise-Reversi
Nützliches und Wissenswertes	3	Editorial
	105	Buchtips
	106	Kontakte: Händler in Ihrer Nähe
	106	Impressum
	107	Bestellkarten

SÄMTLICHE PROGRAMME, die in dieser Ausgabe als Listing mit Dokumentation veröffentlicht wurden, können sie auf Datenträger beziehen. Anforderungskarten am Heftende.

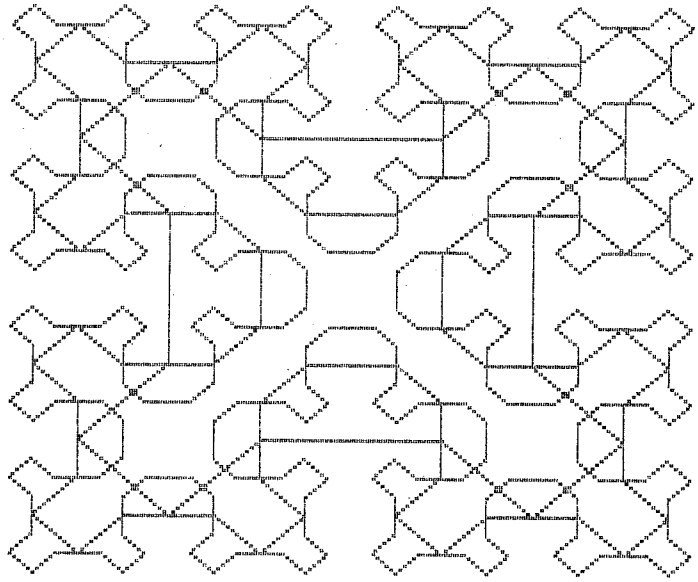
Sierpinski-Kurven

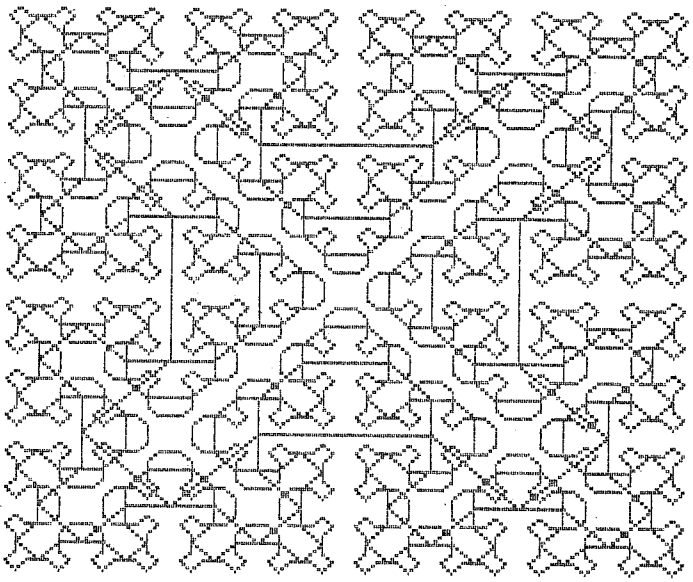
Programmlänge: 1300 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128

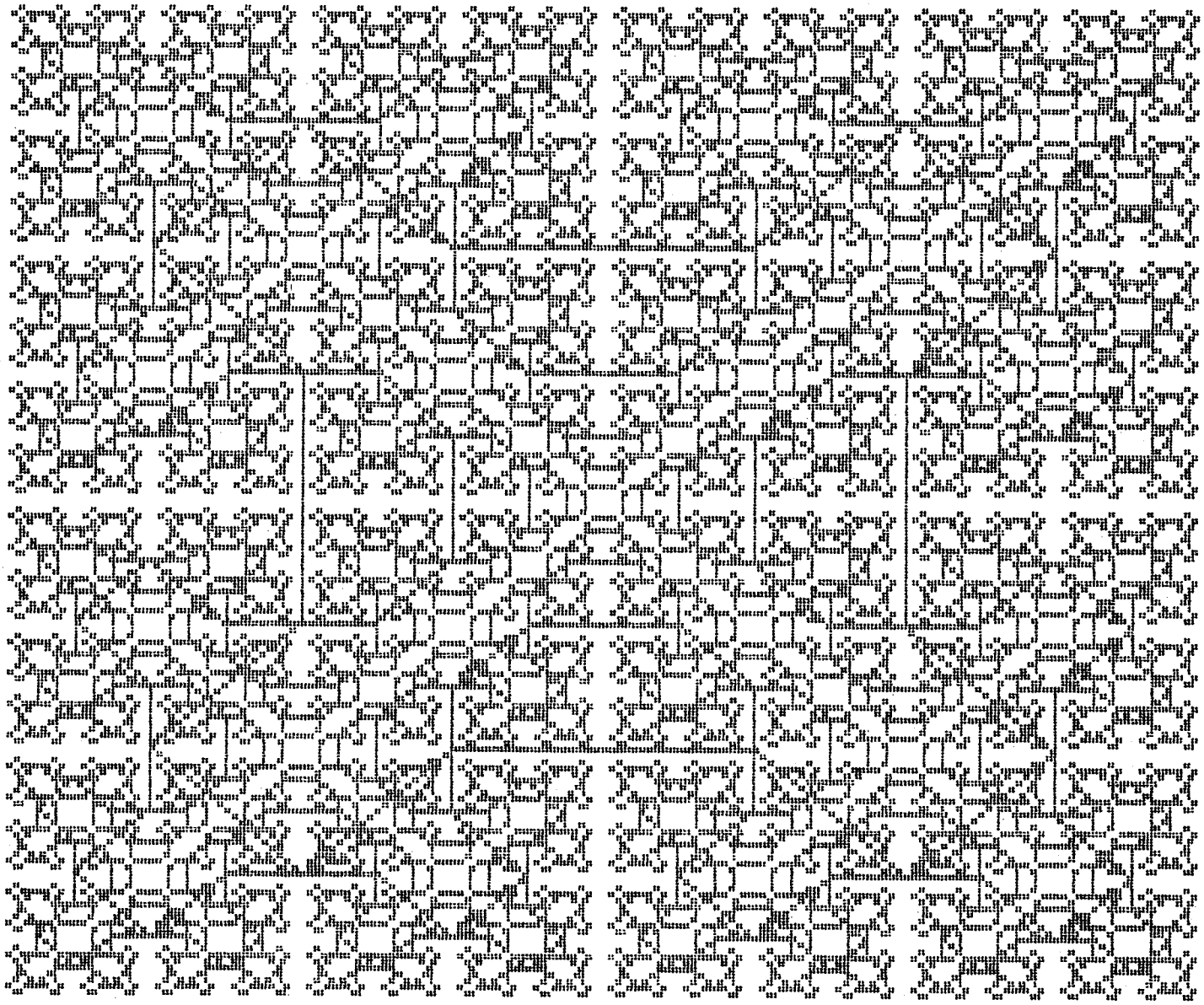
Und hier nun das Gegenstück zu den Hilbert-Kurven: die Sierpinski-Kurven.
Ihre geometrische Struktur ist komplexer und schöner anzusehen als die der Hilbert-Kurven. Die Hardcopy entstand wiederum durch Überlagerung mehrerer Kurven S(i). Dabei steht S(i) für Sierpinski-Kurve i-ter Ordnung.
Das zugrundeliegende Rekursionsschema ist völlig anders geartet als bei Hilbert. Das Blatt S(1) als Einheitskurve zu bestimmen bringt keinen Erfolg. Der wesentliche Unterschied zu Hilbert liegt darin, daß Sierpinski-Kurven geschlossene Gebilde ohne Überschneidungen sind. Also muß das Rekursionsschema eine offene Kurve beschreiben. Die Verbindung der vier Teile darf selbst nicht zum Rekursionsmuster gehören. Das Programm benutzt wiederum die vier Grundprozeduren A, B, C und D, deren Rekursionsschema auch dem Programm entnommen werden kann.

Reiner Kunz

```
  100 PROGRAM "Sierpinski-Kurven"
  110 !
  120 DEF A(I)
  130   IF I>0 THEN
  140     CALL A(I-1):LET X=X+H:LET
 Y=Y-H:PLOT X,Y;
  150     CALL B(I-1):LET X=X+2*H:P
LOT X,Y;
  160     CALL D(I-1):LET X=X+H:LET
 Y=Y+H:PLOT X,Y;
  170     CALL A(I-1)
  180   END IF
  190 END DEF
  200 !
  210 DEF B(I)
  220   IF I>0 THEN
  230     CALL B(I-1):LET X=X-H:LET
 Y=Y-H:PLOT X,Y;
  240     CALL C(I-1):LET Y=Y-2*H:P
LOT X,Y;
  250     CALL A(I-1):LET X=X+H:LET
 Y=Y-H:PLOT X,Y;
  260     CALL B(I-1)
  270   END IF
  280 END DEF
  290 !
  300 DEF C(I)
  310   IF I>0 THEN
  320     CALL C(I-1):LET X=X-H:LET
 Y=Y+H:PLOT X,Y;
  330     CALL D(I-1):LET X=X-2*H:P
LOT X,Y;
  340     CALL B(I-1):LET X=X-H:LET
 Y=Y-H:PLOT X,Y;
  350     CALL C(I-1)
  360   END IF
  370 END DEF
  380 !
  390 DEF D(I)
  400   IF I>0 THEN
  410     CALL D(I-1):LET X=X+H:LET
 Y=Y+H:PLOT X,Y;
  420     CALL A(I-1):LET Y=Y+2*H:P
LOT X,Y;
  430     CALL C(I-1):LET X=X-H:LET
 Y=Y+H:PLOT X,Y;
  440     CALL D(I-1)
  450   END IF
  460 END DEF
  470 !
  480 !
  490 GRAPHICS HIRES 2
  500 !
  510 LET N=4 ! Anzahl der zu zeich
nenden Kurven
  520 LET H0=640 ! Einheitslänge
  530 LET I=0 ! Zählvariable
  540 LET H=INT(H0/4)
  550 LET X0=2*H
  560 LET Y0=3*H
  570 !
  580 DO
  590   LET I=I+1:LET X0=X0-H
  600   LET H=INT(H/2):LET Y0=Y0+H
  610   LET X=X0+320:LET Y=Y0
  620   SET BEAM OFF:PLOT X,Y;
  630   CALL A(I):LET X=X+H:LET Y=Y
-H:PLOT X,Y;
  640   CALL B(I):LET X=X-H:LET Y=Y
-H:PLOT X,Y;
  650   CALL C(I):LET X=X-H:LET Y=Y
+H:PLOT X,Y;
  660   CALL D(I):LET X=X+H:LET Y=Y
+H:PLOT X,Y;
  670 LOOP UNTIL I=N
  680 !
  690 END
```





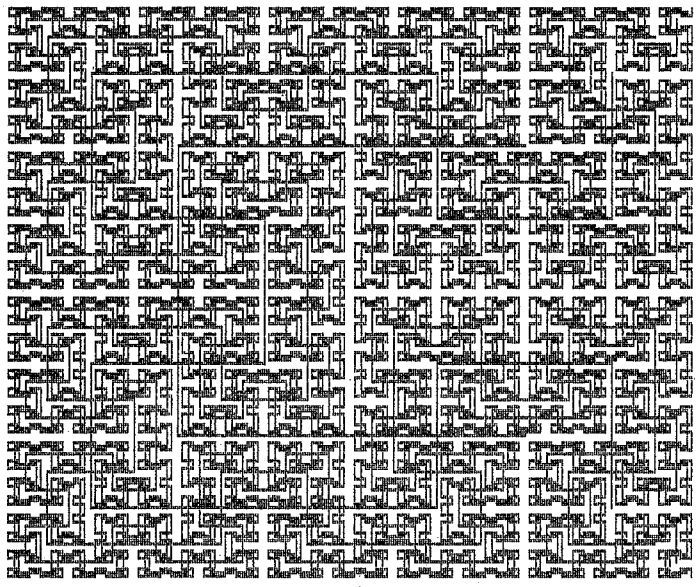


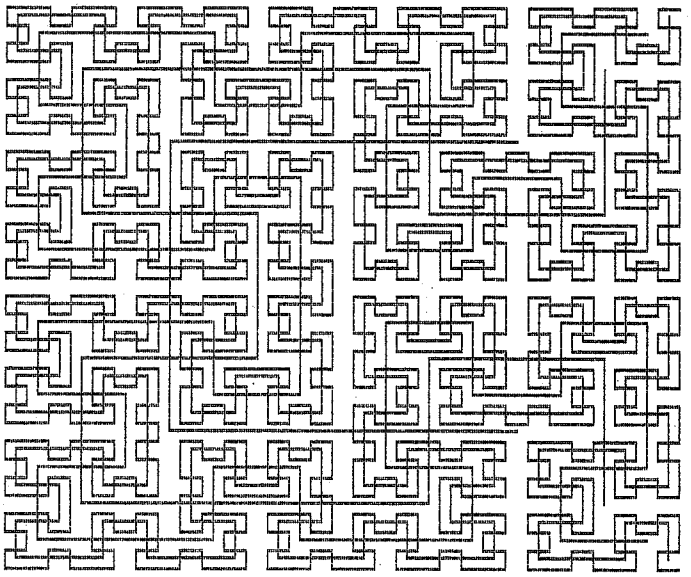
Hilbert-Kurven

Programmlänge: 1160 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128

Hier ist nun eines unserer beiden Garfikprogramme zur Veranschaulichung der Rekursion in BASIC. Die hübschen geometrischen Muster der Hardcopy bestehen aus mehreren übereinander gezeichneten Kurven. Ihre regelmäßige geometrische Struktur läßt auf einen ebenfalls regelmäßigen Algorithmus schließen; eben den rekursiven.
Die Hardcopies zeigen, daß jede Kurve H(i+1) aus jeweils vier Einheiten von H(i) zusammengesetzt ist. Die Größe dieser Einheiten wird halbiert, die Einheiten selbst geeignet gegeneinander rotiert und durch drei gerade Linien verbunden. H(1) kann man sich aus vier Einheiten einer leeren Figur H(0) zusammengesetzt denken, welche mittels dreier gerader Linien untereinander verbunden sind. Übrigens heißt H(i) Hilbert-Kurve der Ordnung i nach ihrem Erfinder D. Hilbert (1891).
Zum Zeichnen der Kurven benötigen wir laut obiger Definition vier Einheitskurven, die entsprechend rotiert und verbunden werden. Diese Prozeduren nennen wir A, B , C und D.
Dabei ist h die Länge der Einheitsstrecke. Gemäß gewisser Rekursionsschemata für A, B, C und D, die man leicht dem Programm entnehmen kann, lassen sich nun die Hilbert-Kurven zeichnen.

Reiner Kunz





```
  100 PROGRAM "Hilbert-Kurven"
  110 !
  120 DEF A(I)
  130   IF I>0 THEN
  140     CALL D(I-1):LET X=X-H:PLO
T X,Y;
  150     CALL A(I-1):LET Y=Y-H:PLO
T X,Y;
  160     CALL A(I-1):LET X=X+H:PLO
T X,Y;
  170     CALL B(I-1)
  180   END IF
  190 END DEF
  200 !
  210 DEF B(I)
  220   IF I>0 THEN
  230     CALL C(I-1):LET Y=Y+H:PLO
T X,Y;
  240     CALL B(I-1):LET X=X+H:PLO
T X,Y;
  250     CALL B(I-1):LET Y=Y-H:PLO
T X,Y;
  260     CALL A(I-1)
  270   END IF
  280 END DEF
  290 !
```

```
  300 DEF C(I)
  310   IF I>0 THEN
  320     CALL B(I-1):LET X=X+H:PLO
T X,Y;
  330     CALL C(I-1):LET Y=Y+H:PLO
T X,Y;
  340     CALL C(I-1):LET X=X-H:PLO
T X,Y;
  350     CALL D(I-1)
  360   END IF
  370 END DEF
  380 !
  390 DEF D(I)
  400   IF I>0 THEN
  410     CALL A(I-1):LET Y=Y-H:PLO
T X,Y;
  420     CALL D(I-1):LET X=X-H:PLO
T X,Y;
  430     CALL D(I-1):LET Y=Y+H:PLO
T X,Y;
  440     CALL C(I-1)
  450   END IF
  460 END DEF
  470 !
  480 !
  490 GRAPHICS HIRES 2
  500 !
  510 LET N=5 ! Anzahl der zu zeich
nenden Kurven
  520 LET I=0 ! Zählvariable
  530 LET H=640 ! Einheitslänge
  540 LET X0=INT(H/2)+H
  550 LET Y0=INT(H/2)
  560 !
  570 DO
  580   LET I=I+1
  590   LET H=INT(H/2)
  600   LET X0=X0+INT(H/2)
  610   LET Y0=Y0+INT(H/2)
  620   LET X=X0-320:LET Y=Y0
  630   SET BEAM OFF:PLOT X,Y;
  640   CALL A(I)
  650 LOOP UNTIL I=N
  660 !
  670 END
```

Vokabeltrainer

Rechner: Enterprise 64/128
Programmname: Voc 1.st
Programmlänge: ca. 16 000 Bytes
Programmiersprache: BASIC
Zubehör: Drucker

Verständigungsschwierigkeiten im Ausland können Sie nun getrost vergessen: Mit diesem Programm können Sie schon zu Hause "Trockenübungen" machen und sich den benötigten Sprachschatz aneignen.

Anzahl der Vokabeln

Die Menge der maximal im Speicher befindlichen Vokabeln kann in Zeile 220 festgelegt werden (AN=...; Default ist 200). Falls das Programm in der Programmebene null liegt (48 kByte zur Verfügung), so kann es ca. 200 Datensätze verwalten. Sind mehr Vokabeln erwünscht, muß man sie in einzelne Lektionen aufteilen.

Starten des Programms

Nach dem Starten benötigt das Programm einige Zeit zur Dimensionierung sowie Initialisierung der Variablen und Arrays; zudem wird auch die Titelgrafik erstellt. Ist dies geschehen, so ertönt ein Klingelzeichen.

Auswahl

Nun befinden Sie sich im Menü: Mit dem Cursorstick können Sie den Zeiger bewegen, durch Drücken von <SPACE> wird das entsprechende Unterprogramm aufgerufen.

Vokabeln eingeben

Hier können Sie den Vokabelstamm ergänzen (vorhandene Vokabeln werden nicht gelöscht); dabei sind folgende "Absicherungen" zu beachten:

* Bei "Vokabel:" darf kein "leerer" String (nur <ENTER>) eingegeben werden, da dies nicht sinnvoll ist (im Gegensatz zu "Bed.:").
* Es muß mindestens eine Bedeutung

angegeben werden, ansonsten wird die Abfrage der Übersetzungen wiederholt.
* Mit "e" ist es möglich, die Eingabe abzubrechen und ins Menü zurückzukehren. Dabei werden die zu diesem Datensatz bereits gemachten Angaben ignoriert.

Nach jedem Datensatz erfolgt eine Abfrage, ob die eingegebenen Daten korrekt sind; ist dies der Fall, so bietet das Programm mit "Weitere Eingaben (j/n)" die Möglichkeit, diese Funktion "regulär" zu verlassen. Nach dem Sortieren verzweigt das Programm ins Menü.

Daten bearbeiten

Unter diesem Menüpunkt können die gespeicherten Daten verändert bzw. ausgedruckt werden. Zuerst müssen Sie die Nummer des Datensatzes, den Sie bearbeiten wollen, eingeben (dies ist nur als Erleichterung gedacht, damit Sie z.B. für "Maier" nicht bei "Adam und Eva" suchen müssen; die genaue Nummer kann evtl. auch über eine der Suchroutinen abgefragt werden). Ist das geschehen, so bieten sich Ihnen folgende Alternativen:
* "+","-" (vorwärts,rückwärts)
  Hiermit lassen Sie sich nächsten bzw. den vorhergehenden Datensatz zeigen (ist dies nicht möglich, so wird der Befehl ignoriert).
* "d" (Drucken)
  Durch Tippen von "d" drucken Sie die aktuelle Vokabel sowie ihre Bedeutungen aus (siehe auch "Datei ausdrucken").
* "ä" (Ändern)
  Mit diesem Befehl können Sie die angezeigten Daten verändern, wobei die bei "Vokabeln eingeben" beschriebenen Einschränkungen zu beachten sind. Anschließend werden die Vokabeln sortiert.
* "l" (Löschen)
  Falls Sie sich vertippt haben sollten, können Sie dies bei der "Sicherheitsabfrage" zurücknehmen: Nur wenn Sie Ihre Wahl bestätigen, wird der Datensatz gelöscht. Auch hier wird der Datensatz neu sortiert.
* "e" (Ende)
  Daraufhin befinden Sie sich wieder im Hauptmenü.

Datei ausdrucken

Diese Funktion gestattet es Ihnen, alle Vokabeln auszudrucken. Dabei können Sie noch eine Überschrift wählen (z.B. "Lektion 11"; "e" bricht das Unterprogramm ab). Nach Beendigung des Drucks verzweigt das Programm wieder ins Hauptmenü.

Vokabeln abfragen

Nun müssen Sie angeben, ob die Abfrage alphabetisch oder zufallsgesteuert geschehen soll; anschließend müssen Sie sich noch zwischen Prüfen der Vokabeln und Abfrage der Bedeutungen entscheiden. Das Auswählen der Vokabeln richtet sich nach folgenden Regeln:
* Keine Vokabel bzw. Bedeutung wird öfter als dreimal abgefragt.
* Wurde eine richtige Antwort gegeben, so wird auch diese Frage nicht mehr gestellt.

Bei jeder falschen Eingabe verringert sich der Punktestand um eins. Gibt es keine Vokabeln mehr, für die die beiden obigen Regeln nicht zutreffen, so ist die Abfrage beendet (dies kann auch durch "e" bewirkt werden).
Im Anschluß daran erfolgt eine Ergebnisausgabe, die dem Schüler gestattet, seine Leistungen einzuschätzen; nach einem Tastendruck befinden Sie sich wieder im Hauptmenü.

Lektion speichern

Hiermit speichern Sie die Vokabeln und ihre Bedeutungen (nicht das Programm!) ab. Zu diesem Zweck müssen Sie noch einen Dateinamen angeben, unter dem die Lektion abgelegt werden soll ("e" führt zurück zum Hauptmenü). Nun erfolgt ein Rücksprung ins Menü.

Lektion einladen

Dies ist die umgekehrte Funktion: Hier müssen Sie den Namen der Datei angeben, die eingeladen werden

soll. Nach dem Einlesen der Daten wird wieder das Hauptmenü aufgerufen.

Programmende

Diesen Punkt sollten Sie erst anwählen, wenn Sie Ihre Daten abgespeichert haben (das Programm initialisiert zu Beginn alle Datensätze)!!!

Thomas Müller

```
  100 !Voc 1.st
  110 !für Chip Special
  120 !von Thomas Müller
  130 !
  140 GOTO 180
  150 CALL MENUE
  160 END
  170 !
  180 !**** Initialisierung ****
  190 !
  200 CLEAR TEXT:SET STATUS OFF
  210 OPTION ANGLE DEGREES
  220 LET AN=200
  230 DIM PO(AN+30),T(6),P(3),A(AN)
  240 STRING VOC$(AN,3)*25,UP$(6)*20,G$(4)*40,H$(3)*80,C$(2)*25,L$(1)*40
  250 LET X,Y,AW,I,M,D,EF,FF,F1,F2,AK,TH=0
  260 LET AB,A1,A2,PU,GF,IN,RI,H1,H2,H3=0
  270 LET A$,G$(1),G$(2),C$(1),C$(2),L$(1)=""
  280 LET G$(3)="%####":LET G$(4)="%####"
  290 LET IK=130:LET AD=8:LET R=400:LET D=1
  300 FOR I=0 TO 6
  310   READ UP$(I)
  320   LET T(I)=22-IP(LEN(UP$(I))/2)
  330 NEXT I
  340 FOR I=1 TO 36
  350   LET G$(1)=G$(1)&CHR$(153):LET L$(1)=L$(1)&CHR$(32)
  360 NEXT I
  370 FOR I=1 TO 25
  380   LET G$(2)=G$(2)&CHR$(46):LET C$(1)=C$(1)&CHR$(0):LET C$(2)=C$(2)&CHR$(255)
  390 NEXT I
  400 LET P(0)=31:LET P(1)=15:LET P(2)=7:LET P(3)=3
  410 FOR I=0 TO AN
  420   LET PO(I+30)=I
  430   LET VOC$(I,0)=C$(2)
  440 NEXT I
  450 !--- Daten für UP$ ---
  460 DATA "Vokabeln eingeben"
  470 DATA "Daten bearbeiten"
  480 DATA "Datei ausdrucken"
  490 DATA "Vokabeln abfragen"
  500 DATA "Lektion speichern"
  510 DATA "Lektion einladen"
  520 DATA "Programm beenden"
  530 !--- window 1 ---
  540 SET VIDEO MODE 1
  550 SET VIDEO COLOUR 2
  560 SET VIDEO X 42
  570 SET VIDEO Y 25
  580 OPEN #1:"video:"
  590 CLEAR #1
  600 SET #1:PALETTE BLACK,BLUE,WHITE,RED,0,0,0,0
  610 !--- window 2 ---
  620 SET VIDEO MODE 1
  630 SET VIDEO COLOUR 1
  640 SET VIDEO X 42
  650 SET VIDEO Y 25
  660 OPEN #2:"video:"
  670 CLEAR #2
  680 SET #2:PALETTE BLACK,BLUE,RGB(.7,.2,.9),RED,0,0,0
  690 SET #2:CHARACTER 150,255,248,224,192,192,192,224,248,255
  700 SET #2:CHARACTER 151,255,31,7,3,3,3,7,31,255
  710 SET #2:CHARACTER 152,192,224,240,248,252,248,240,224,192
  720 SET #2:CURSOR CHARACTER 32
  730 !--- window 3 ---
  740 SET VIDEO MODE 2
  750 SET VIDEO COLOUR 0
  760 SET VIDEO X 42
  770 SET VIDEO Y 1
  780 OPEN #3:"video:"
  790 SET #3:CURSOR CHARACTER 32
  800 SET #3:CHARACTER 64,60,60,60,60,60,60,60,60,60
  810 !--- Textfenster ---
  820 SET #102:CHARACTER 153,0,0,0,255,255,255,0,0,0
  830 SET #102:CHARACTER 154,0,0,0,31,31,31,24,24,24
  840 SET #102:CHARACTER 155,0,0,0,248,248,248,24,24,24
  850 SET #102:CHARACTER 156,24,24,24,31,31,31,0,0,0
  860 SET #102:CHARACTER 157,24,24,24,248,248,248,0,0,0
  870 SET #102:CHARACTER 158,24,24,
```

```
24,24,24,24,24,24,24
  880 SET #102:PALETTE BLACK,RGB(.6
,.6,.7),RGB(.7,.2,.9),RED,0,0,0,0
  890 !--- window 4 ---
  900 SET VIDEO MODE 0
  910 SET VIDEO COLOUR 0
  920 SET VIDEO X 42
  930 SET VIDEO Y 23
  940 OPEN #4:"video:"
  950 SET #4:CURSOR CHARACTER 32
  960 CLEAR #4
  970 !--- window 5 ---
  980 SET VIDEO MODE 1
  990 SET VIDEO COLOUR 1
 1000 SET VIDEO X 42
 1010 SET VIDEO Y 3
 1020 OPEN #5:"video:"
 1030 SET #5:CURSOR CHARACTER 32
 1040 CLEAR #5
 1050 SET #5:PALETTE BLACK,RGB(.6,.
6,.7),YELLOW,RED,0,0,0,0
 1060 !--- Titelbild ---
 1070 LET I=0
 1080 LET X=1.5*R*COS(AW):LET Y=R*S
IN(AW)
 1090 PLOT #1:672+X,450+Y;
 1100 LET D=1-(1-D)*ABS(IK)/360:LET
 AD=AD*ABS(IK)/360
 1110 LET R=R*D+AD:LET AW=AW+IK
 1120 IF AW>360 THEN LET AW=AW-IP(A
W/360)*360
 1130 LET I=I+1
 1140 IF I<40 THEN 1080
 1150 PING
 1160 PING
 1170 DISPLAY #1:AT 1 FROM 1 TO 25
 1180 SET #1:BEAM OFF
 1190 SET #1:INK 2
 1200 PLOT #1:580,540,
 1210 PRINT #1:"Voc"
 1220 PLOT #1:560,440,
 1230 PRINT #1:"1.st"
 1240 CALL WAIT
 1250 GOTO 150
 1260 !
 1270 !**** Menü ****
 1280 !
 1290 DEF MENUE
 1300   SET #1:INK 2
 1310   CLEAR #1
 1320   DISPLAY #1:AT 1 FROM 1 TO 2
 1330   PLOT #1:400,899,
 1340   PRINT #1:"Voc 1.st"
 1350   FOR I=4 TO 19
 1360     PLOT #2:0,I*36-4,
 1370     PRINT #2:CHR$(150)
 1380     PLOT #2:1312,I*36-4,
 1390     PRINT #2:CHR$(151);
 1400   NEXT I
 1410   SET #2:LINE STYLE 2
 1420   FOR I=4 TO 32 STEP 4
 1430     PLOT #2:0,680+I;1343,680+
I
 1440     PLOT #2:0,72+I;1343,72+I
 1450   NEXT I
 1460   PLOT #2:0,640,
 1470   FOR I=0 TO 6
 1480     PRINT #2:TAB(T(I));UP$(I)

 1490     PRINT #2:
 1500   NEXT I
 1510   DISPLAY #2:AT 3 FROM 3 TO 2
5
 1520   !--- Auswahl ---
 1530   LET I=0:LET G=0
 1540   SET #2:INK 0
 1550   PLOT #2:T(G)*32-80,640-G*72
,
 1560   PRINT #2:CHR$(159);
 1570   SET #2:INK 2
 1580   PLOT #2:T(I)*32-80,640-I*72
,
 1590   PRINT #2:CHR$(152);
 1600   CALL WAIT
 1610   LET AS=ORD(A$)
 1620   IF AS=176 THEN LET G=I:LET
I=I-SGN(I):GOTO 1540
 1630   IF AS=180 THEN LET G=I:LET
I=I+SGN(6-I):GOTO 1540
 1640   IF AS<>32 THEN 1600
 1650   LET AS=I
 1660   SET #2:INK 0
 1670   PLOT #2:T(I)*32-80,640-I*72
,
 1680   PRINT #2:CHR$(159);
 1690   !--- window 3 setzen ---
 1700   CLEAR #3
 1710   DISPLAY #3:AT 25 FROM 1 TO
1
 1720   !--- Aufrufe ---
 1730   SELECT CASE AS
 1740     !--- Vokabeln eingeben --
-
 1750   CASE 0
 1760     CALL EINGABE
 1770     CALL SORT_2
 1780     !--- Daten bearbeiten ---
 1790   CASE 1
 1800     CALL CHANGE_1
 1810     IF FF=1 THEN 2050
 1820     CALL CHANGE_2
 1830     !--- Datei ausdrucken ---

 1840   CASE 2
```

```
 1850      CALL DRUCK
 1860      !--- Vokabeln abfragen --
 1870    CASE 3
 1880      CALL MENUE2
 1890      IF FF=1 THEN 2050
 1900      CALL ASK
 1910      !--- Speichern ---
 1920    CASE 4
 1930      CALL SAVE
 1940      !--- Einladen ---
 1950    CASE 5
 1960      CALL LOAD
 1970      !--- Ende ---
 1980    CASE 6
 1990      CLOSE #1
 2000      CLOSE #2
 2010      CLOSE #3
 2020      TEXT 40
 2030      GOTO 2060
 2040    END SELECT
 2050    GOTO 1510
 2060 END DEF
 2070 !
 2080 !**** Eingabe ****
 2090 !
 2100 DEF EINGABE
 2110    CLEAR TEXT
 2120    SET #102:INK 1
 2130    DISPLAY #4:AT 3 FROM 1 TO 2
2
 2140    IF AK+1>AN THEN CALL FULL:L
ET FF=1:GOTO 2690
 2150    PRINT AT 4,15:"Eingabe"
 2160    PRINT AT 5,15:"======="
 2170    PRINT AT 7,0:CHR$(154);G$(1
);CHR$(155)
 2180    PRINT AT 8,0:CHR$(158);"Vok
abel:    ";TAB(38);CHR$(158)
 2190    PRINT AT 9,0:CHR$(156);G$(1
);CHR$(157)
 2200    FOR I=0 TO 2
 2210      PRINT AT 12+I*3,0:CHR$(15
4);G$(1);CHR$(155)
 2220      PRINT AT 13+I*3,0:CHR$(15
8);STR$(I+1);".Bed.:    ";TAB(38);C
HR$(158)
 2230      PRINT AT 14+I*3,0:CHR$(15
6);G$(1);CHR$(157)
 2240    NEXT I
 2250    DISPLAY #102:AT 3 FROM 3 TO
24
 2260    LET AK=AK+1:LET TH=TH+1
 2270    PRINT AT 8,13:G$(2);
 2280    FOR I=0 TO 2
 2290      PRINT AT 13+I*3,13:G$(2);
 2300    NEXT I
 2310    INPUT AT 8,13,PROMPT "":H$(
1)
 2320    IF H$(1)=G$(2)&CHR$(158) TH
EN 2310
 2330    LET H1=0:LET H2=AK
 2340    CALL TEST
 2350    IF EF=1 THEN 2660
 2360    LET F1=0
 2370    FOR I=1 TO 3
 2380      LET F2=0
 2390      INPUT AT 10+I*3,13,PROMPT
"":H$(1)
 2400      LET H1=I:LET H2=AK
 2410      CALL TEST
 2420      IF EF=1 THEN EXIT FOR
 2430      LET F1=F1+F2
 2440    NEXT I
 2450    IF F1=3 THEN 2360
 2460    IF EF=1 THEN 2660
 2470    PRINT AT 22,7:"Alle Eingabe
n korrekt (j/n)?"
 2480    CALL WAIT
 2490    IF A$="n" THEN
 2500      PRINT AT 22,0:L$(1)
 2510      GOTO 2310
 2520    END IF
 2530    IF A$<>"j" THEN 2480
 2540    PRINT AT 22,36:"j"
 2550    PRINT AT 24,10:"Weitere Ein
gaben (j/n)?";
 2560    CALL WAIT
 2570    IF A$="j" THEN
 2580      PRINT AT 3,0:
 2590      PRINT AT 22,0:L$(1)
 2600      PRINT AT 24,0:L$(1);
 2610      PRINT AT 3,0:
 2620      GOTO 2260
 2630    END IF
 2640    IF A$<>"n" THEN 2560
 2650    GOTO 2690
 2660    !--- Eingabe-Abbruch ---
 2670    LET TH=TH-1:LET AK=AK-1
 2680    CALL WAIT
 2690 END DEF
 2700 !
 2710 !**** Daten bearbeiten ****
 2720 !
 2730 DEF CHANGE_1
 2740    LET FF=0:CLEAR TEXT
 2750    DISPLAY #102:AT 3 FROM 3 TO
24
 2760    IF AK=0 THEN CALL EMPTY:LET
FF=1:GOTO 2820
 2770    PRINT AT 12,10:"Welcher Dat
ensatz soll"
 2780    INPUT AT 14,10,PROMPT "bear
beitet werden? ":H$(1)
 2790    IF H$(1)="e" THEN LET FF=1:
```

```
GOTO 2820
 2800    LET D=VAL(H$(1))
 2810    IF D<1 OR D>AK THEN 2780
 2820 END DEF
 2830 !
 2840 !**** UP Daten bearbeiten ***
*
 2850 !
 2860 DEF CHANGE_2
 2870    DISPLAY #3:AT 25 FROM 1 TO
1
 2880    PRINT #3:"     § + vorwärts
 § - rückwärts § D Drucken § Ä ände
rn § L löschen § E Ende §";
 2890    CLEAR TEXT
 2900    DISPLAY #4:AT 3 FROM 1 TO 2
2
 2910    PRINT AT 4,0:"Datensatz Nr.
: "
 2920    PRINT AT 7,0:CHR$(154);G$(1
);CHR$(155)
 2930    PRINT AT 8,0:CHR$(158);"Vok
abel:    ";TAB(38);CHR$(158)
 2940    PRINT AT 9,0:CHR$(156);G$(1
);CHR$(157)
 2950    FOR I=0 TO 2
 2960      PRINT AT 12+I*3,0:CHR$(15
4);G$(1);CHR$(155)
 2970      PRINT AT 13+I*3,0:CHR$(15
8);STR$(I+1);".Bed.:    ";TAB(38);C
HR$(158)
 2980      PRINT AT 14+I*3,0:CHR$(15
6);G$(1);CHR$(157)
 2990    NEXT I
 3000    DISPLAY #102:AT 3 FROM 3 TO
 24
 3010    PRINT AT 8,13:VOC$(PO(D+30)
,0);
 3020    FOR I=0 TO 2
 3030      PRINT AT 13+I*3,13:VOC$(P
O(D+30),I+1);
 3040    NEXT I
 3050    PRINT AT 4,16:D;
 3060    CALL WAIT
 3070    SELECT CASE ORD(A$)
 3080      !--- vorwärts ---
 3090    CASE 43
 3100      LET D=D+SGN(AK-D)
 3110      GOTO 3010
 3120      !--- rückwärts ---
 3130    CASE 45
 3140      LET D=D-SGN(D-1)
 3150      GOTO 3010
 3160      !--- Drucken ---
 3170    CASE 100
 3180      LPRINT
 3190      LPRINT VOC$(PO(30+D),0);"
          ";VOC$(PO(30+D),1)
 3200      FOR I=2 TO 3
 3210        LPRINT L$(1);VOC$(PO(30
+D),I)
 3220      NEXT I
 3230      GOTO 3060
 3240      !--- Ändern ---
 3250    CASE 123
 3260      INPUT AT 8,13,PROMPT "":H
$(0)
 3270      IF H$(0)=G$(2)&CHR$(158)
THEN 3260
 3280      LET H1=0:LET H2=D:LET H3=
0
 3290      CALL TEST_2
 3300      IF EF=1 THEN 3010
 3310      LET F1=0
 3320      FOR I=1 TO 3
 3330        LET F2=0
 3340        INPUT AT 10+I*3,13,PROM
PT "":H$(I)
 3350        LET H1=I:LET H2=D:LET H
3=I
 3360        CALL TEST_2
 3370        IF EF=1 THEN EXIT FOR
 3380        LET F1=F1+F2
 3390      NEXT I
 3400      IF EF=1 THEN 3010
 3410      IF F1=3 THEN 3330
 3420      PRINT AT 21,7:"Alle Einga
ben korrekt (j/n)?"
 3430      CALL WAIT
 3440      IF A$="j" THEN
 3450        FOR I=0 TO 3
 3460          LET VOC$(PO(30+D),I)=
H$(I)(1:25)
 3470        NEXT I
 3480        PRINT AT 21,0:L$(1)
 3490        CALL SORT_2
 3500        GOTO 2890
 3510      END IF
 3520      IF A$<>"n" THEN 3430
 3530      PRINT AT 21,0:L$(1)
 3540      GOTO 3260
 3550      !--- Löschen ---
 3560    CASE 108
 3570      PRINT AT 22,0:
 3580      PRINT AT 22,3:"Löschen: S
ind Sie sicher (j/n)?";
 3590      CALL WAIT
 3600      IF A$="n" THEN 3260
 3610      IF A$<>"j" THEN 3590
 3620      FOR I=0 TO 3
 3630        LET VOC$(PO(30+D),I)=C$
(2)
 3640      NEXT I
 3650      LET TH=TH-1
```

```
 3660      IF TH=0 THEN LET AK=0:CAL
L EMPTY:GOTO 3760
 3670      CALL SORT_2
 3680      IF D>AK THEN LET D=AK
 3690      IF D<1 THEN LET D=1
 3700      GOTO 2890
 3710      !--- Ende ---
 3720    CASE 101
 3730      GOTO 3770
 3740    CASE ELSE
 3750      GOTO 3060
 3760    END SELECT
 3770 END DEF
 3780 !
 3790 !**** Shell-Sort UP ****
 3800 !
 3810 DEF SORT_2
 3820    CLEAR TEXT
 3830    PRINT AT 10,10:"Datenstamm
wird sortiert"
 3840    PRINT AT 11,10:"==========
============="
 3850    FOR M=0 TO 3
 3860      CALL SORT_UP
 3870    NEXT M
 3880    !--- Last Sort ---
 3890    FOR I=32 TO AK+30
 3900      LET PO(30)=PO(I):LET L=I-
1
 3910      IF VOC$(PO(L),0)>VOC$(PO(
30),0) THEN
 3920        LET PO(L+1)=PO(L):LET L
=L-1:GOTO 3910
 3930      END IF
 3940      LET PO(L+1)=PO(30)
 3950    NEXT I
 3960    LET AK=TH
 3970    CLEAR TEXT
 3980 END DEF
 3990 !--- Sort - Unterprogramm ---

 4000 DEF SORT_UP
 4010    IF 2*P(M)>AK THEN 4100
 4020    FOR I=31 TO P(M)+30
 4030      FOR I2=I+P(M) TO AK+30-P(
M) STEP P(M)
 4040        LET PO(I-P(M))=PO(I2):L
ET L=I2-P(M)
 4050        IF VOC$(PO(L),0)<=VOC$(
PO(I-P(M)),0) THEN 4070
 4060        LET PO(L+P(M))=PO(L):LE
T L=L-P(M):GOTO 4050
 4070        LET PO(L+P(M))=PO(I-P(M
))
 4080      NEXT I2
 4090    NEXT I
 4100 END DEF
 4110 !
 4120 !**** Eingabe-Test ****
 4130 !
 4140 DEF TEST
 4150    LET EF=0
 4160    IF H$(1)(1:25)="e"&G$(2)(1:
24) THEN LET EF=1:GOTO 4200
 4170    IF H$(1)=G$(2)&CHR$(158) TH
EN LET F2=1
 4180    LET VOC$(H2,H1)=H$(1)(1:25)

 4190    GOTO 4230
 4200    FOR I=0 TO 3
 4210      LET VOC$(H2,I)=C$(2)
 4220    NEXT I
 4230 END DEF
 4240 !
 4250 !**** Eingabe 2 ****
 4260 !
 4270 DEF TEST_2
 4280    LET EF=0
 4290    IF H$(H3)(1:25)="e"&G$(2)(1
:24) THEN LET EF=1:GOTO 4310
 4300    IF H$(H3)=G$(2)&CHR$(158) T
HEN LET F2=1
 4310 END DEF
 4320 !
 4330 !**** Warteschleife ****
 4340 !
 4350 DEF WAIT
 4360    LET A$=INKEY$
 4370    IF A$="" THEN 4360
 4380 END DEF
 4390 !
 4400 !**** Keine Daten vorhanden *
***
 4410 !
 4420 DEF EMPTY
 4430    CLEAR TEXT
 4440    PRINT AT 10,14:"Datei leer:
"
 4450    PRINT AT 12,10:"Keine Daten
 verfügbar!"
 4460    CALL WAIT
 4470 END DEF
 4480 !
 4490 !**** Abfragemenü 1 ****
 4500 !
 4510 DEF MENUE2
 4520    LET FF=0
 4530    IF AK=0 THEN LET FF=1:GOTO
5590
 4540    CLEAR TEXT
 4550    DISPLAY #102:AT 3 FROM 3 TO
 24
 4560    PRINT AT 10,7:"1 <= Alphabe
tische Abfrage"
```

```
 4570   PRINT AT 12,7:"2 <= Zufalls
gesteuert"
 4580   CALL WAIT
 4590   LET A1=VAL(A$)
 4600   IF A1<>1 AND A1<>2 THEN 458
0
 4610   CLEAR TEXT
 4620   PRINT AT 10,13:"1 <= Vokabe
ln"
 4630   PRINT AT 12,13:"2 <= Bedeut
ung"
 4640   CALL WAIT
 4650   LET A2=VAL(A$)
 4660   IF A2<>1 AND A2<>2 THEN 464
0
 4670 END DEF
 4680 !
 4690 !**** Abfragen ****
 4700 !
 4710 DEF ASK
 4720   FOR I=1 TO AK
 4730     LET A(I)=0
 4740   NEXT I
 4750   CALL DESK
 4760   LET PU=3*AK:LET D=0:LET GF=
0:LET RI=0:LET IN=0
 4770   PRINT AT 3,13,USING G$(3):P
U
 4780   PRINT AT 10,32,USING G$(4):
AK;
 4790   !--- Datensatz wählen ---
 4800   SELECT CASE A1
 4810     !--- alphabetisch ---
 4820   CASE 1
 4830     LET D=D+1
 4840     IF D>AK THEN LET D=1
 4850     !--- zufällig ---
 4860   CASE 2
 4870     LET D=RND(AK)+1
 4880   END SELECT
 4890   IF A(D)=3 THEN 4800
 4900   !--- abfragen ---
 4910   SELECT CASE A2
 4920     !--- Vokabel ---
 4930   CASE 1
 4940     FOR I=1 TO 3
 4950       PRINT AT 13+I*3,13:G$(2
);
 4960     NEXT I
 4970     PRINT AT 6,13:VOC$(PO(30+
D),0);
 4980     PRINT AT 16,13:G$(2);
 4990     INPUT AT 16,13,PROMPT "":
H$(1)
 5000     LET H3=1:LET EF,F2=0
 5010     CALL TEST_2
 5020     IF EF=1 THEN 5500
 5030     IF F2=1 THEN 4990
 5040     LET FF=1:LET H$(1)=H$(1)(
1:25)
 5050     FOR I=1 TO 3
 5060       IF VOC$(PO(30+D),I)=H$(
1) THEN LET FF=0:EXIT FOR
 5070     NEXT I
 5080     CALL COUNT
 5090     IF FF=0 THEN 5140
 5100     FOR I=1 TO 3
 5110       PRINT AT 13+I*3,13:VOC$
(PO(30+D),I);
 5120     NEXT I
 5130     CALL WAIT
 5140     !--- Bedeutungen ---
 5150   CASE 2
 5160     PRINT AT 6,13:G$(2);
 5170     FOR I=1 TO 3
 5180       PRINT AT 13+I*3,13:VOC$
(PO(30+D),I);
 5190     NEXT I
 5200     INPUT AT 6,13,PROMPT "":H
$(1)
 5210     LET H3=1:LET EF,F2=0
 5220     CALL TEST_2
 5230     IF EF=1 THEN 5500
 5240     IF F2=1 THEN 5200
 5250     IF VOC$(PO(30+D),0)=H$(1)
(1:25) THEN
 5260       LET FF=0
 5270     ELSE
 5280       LET FF=1
 5290     END IF
 5300     CALL COUNT
 5310     IF FF=0 THEN 5340
 5320     PRINT AT 6,13:VOC$(PO(30+
D),0);
 5330     CALL WAIT
 5340   END SELECT
 5350   !--- Balken ---
 5360   LET H1=IP(445/AK*GF)
 5370   IF H1<4 THEN 5400
 5380   SET #5:INK 3
 5390   PLOT #5:H1,48;H1,12,H1-4,12
,PAINT
 5400   LET H2=IP(1336/AK*RI)
 5410   IF H2<4 THEN 5450
 5420   SET #5:INK 2
 5430   PLOT #5:H2,96;H2,56,H2-4,60
,PAINT
 5440   !--- Zwischenergebnisse ---

 5450   PRINT AT 3,13,USING G$(3):P
U;
 5460   PRINT AT 10,10,USING G$(4):
RI;
 5470   PRINT AT 10,32,USING G$(4):
```

```
AK-RI;
 5480    IF IN>=AK*3 THEN 5500
 5490    GOTO 4790
 5500    CLEAR TEXT
 5510    DISPLAY #102:AT 3 FROM 3 TO
 24
 5520    PRINT AT 7,15:"Punkte:";PU
 5530    PRINT AT 9,0:"Vokabeln im S
peicher:";AK
 5540    IF RI+GF=0 THEN 5580
 5550    PRINT AT 11,3:"Abfragen ins
gesamt:";RI+GF
 5560    PRINT AT 13,8:"davon richti
g:";RI;TAB(25);"= ";IP(RI/(RI+GF)*1
00);"%"
 5570    PRINT AT 15,15:"falsch:";GF
;TAB(25);"= ";IP(GF/(RI+GF)*100);"%
"
 5580    CALL WAIT
 5590 END DEF
 5600 !
 5610 !**** Abfrage-Maske ****
 5620 !
 5630 DEF DESK
 5640    DISPLAY #4:AT 3 FROM 1 TO 2
2
 5650    CLEAR TEXT
 5660    PRINT AT 3,5:"Punkte: 00000
"
 5670    PRINT AT 5,0:CHR$(154);G$(1
);CHR$(155)
 5680    PRINT AT 6,0:CHR$(158);"Vok
abel:    ";TAB(38);CHR$(158)
 5690    PRINT AT 7,0:CHR$(156);G$(1
);CHR$(157)
 5700    PRINT AT 8,7:AK;"Vokabeln i
m Speicher"
 5710    PRINT AT 10,0:"Richtig: 000
0     Verbleibend: 0000"
 5720    FOR I=0 TO 2
 5730      PRINT AT 15+I*3,0:CHR$(15
4);G$(1);CHR$(155)
 5740      PRINT AT 16+I*3,0:CHR$(15
8);STR$(I+1);".Bed.:    ";TAB(38);C
HR$(158)
 5750      PRINT AT 17+I*3,0:CHR$(15
6);G$(1);CHR$(157)
 5760    NEXT I
 5770    CLEAR #5
 5780    SET #5:INK 1
 5790    PLOT #5:0,100;1340,100;1340
,8;0,8;0,100
 5800    PLOT #5:0,52;1340,52
 5810    DISPLAY #102:AT 3 FROM 3 TO
 24
 5820    DISPLAY #5:AT 12 FROM 1 TO
3
 5830 END DEF
 5840 !
 5850 !**** Hochzählen ****
 5860 !
 5870 DEF COUNT
 5880    IF FF=1 THEN
 5890      LET PU=PU-1:LET GF=GF+1:L
ET A(D)=A(D)+1
 5900    ELSE
 5910      LET RI=RI+1:LET IN=IN+(2-
A(D)):LET A(D)=3
 5920    END IF
 5930    LET IN=IN+1
 5940 END DEF
 5950 !
 5960 !**** Ausdrucken der ganzen D
atei ****
 5970 !
 5980 DEF DRUCK
 5990    CLEAR TEXT
 6000    DISPLAY #102:AT 3 FROM 3 TO
 24
 6010    PRINT AT 8,15:"Ausdrucken"
 6020    PRINT AT 9,15:"==========="
 6030    INPUT AT 13,5,PROMPT "Übers
chrift: ":H$(1)
 6040    IF H$(1)="e" THEN 6140
 6050    LPRINT
 6060    LPRINT H$(1)
 6070    LPRINT "------------------
---------------------------------
-------"
 6080    LPRINT
 6090    FOR I=1 TO AK
 6100      LPRINT VOC$(PO(30+I),0);"
           ";VOC$(PO(30+I),1)
 6110      LPRINT L$(1);VOC$(PO(30+I
),2)
 6120      LPRINT L$(1);VOC$(PO(30+I
),3)
 6130    NEXT I
 6140 END DEF
 6150 !
 6160 !**** Lektion speichern ****
 6170 !
 6180 DEF SAVE
 6190    CLEAR TEXT
 6200    DISPLAY #102:AT 3 FROM 3 TO
 24
 6210    PRINT AT 9,15:"Abspeichern"
 6220    PRINT AT 10,15:"==========
"
 6230    PRINT AT 14,10:"Dateiname:
    "
 6240    INPUT AT 14,21,PROMPT "":H$
(1)
```

```
 6250   IF H$(1)="e___" THEN 6350
 6260   LET H$(1)=H$(1)(1:4)&"_voc"
 6270   OPEN #8:H$(1) ACCESS OUTPUT
 6280   PRINT #8:AK
 6290   FOR I=1 TO AK
 6300     FOR I2=0 TO 3
 6310       PRINT #8:VOC$(PO(30+I),
I2)
 6320     NEXT I2
 6330   NEXT I
 6340   CLOSE #8
 6350 END DEF
 6360 !
 6370 !**** Lektion einladen ****
 6380 !
 6390 DEF LOAD
 6400   CLEAR TEXT
 6410   DISPLAY #102:AT 3 FROM 3 TO
 24
 6420   PRINT AT 9,15:"Einladen"
 6430   PRINT AT 10,15:"========="
 6440   PRINT AT 14,10:"Dateiname:
      "
 6450   INPUT AT 14,21,PROMPT "":H$
(1)
 6460   IF H$(1)="e___" THEN 6650
 6470   FOR I=1 TO AK
 6480     FOR I2=0 TO 3
 6490       LET VOC$(I,I2)=C$(2)
 6500     NEXT I2
 6510   NEXT I
 6520   FOR I=1 TO AN
 6530     LET PO(30+I)=I
 6540   NEXT I
 6550   LET H$(1)=H$(1)(1:4)&"_voc"
 6560   OPEN #8:H$(1) ACCESS INPUT
 6570   INPUT #8:AK
 6580   FOR I=1 TO AK
 6590     FOR I2=0 TO 3
 6600       INPUT #8:VOC$(I,I2)
 6610     NEXT I2
 6620   NEXT I
 6630   CLOSE #8
 6640   LET TH=AK
 6650 END DEF
```

Integralrechnen

Rechner: Enterprise 64/128
Programmname: Simpson-Integration
Programmlänge: ca. 3300 Bytes
Programmiersprache: BASIC
Zubehör: ---

Die wohl bekannteste Methode für eine numerische Integration ist die Anwendung der Trapezregel: Dabei wird die Fläche in n Streifen, welche von P(i-1) und P(i) nach oben begrenzt sind, unterteilt; daraufhin wird der Bogen P(i-1)P(i) durch die Sehne P(i-1)P(i) ersetzt (das hieraus resultierende Trapez läßt sich leicht berechnen).
Da jedoch eine Parabel im Normalfall eine geringere Abweichung von einem solchen Bogenstück aufweist als eine Gerade (Prisma-Regel), kann man meist durch eine kleinere Anzahl von Feldern (durch Parabelbögen nach oben begrenzt) bereits eine bessere Näherung erzielen.
Auf dieser Tatsache beruht das Simpson-Verfahren: Da eine Parabel mit minimal drei Punkten definiert ist, benötigt man also je zwei Streifen (drei Begrenzungspunkte). Daraus folgt, daß die Gesamtfläche des Integrals in eine gerade Anzahl von Streifen (n) aufgeteilt werden muß. Nun wird n so oft verdoppelt, bis die vom Benutzer gewünschte Genauigkeit erreicht ist.
Die Formel hierfür lautet:

$$(f(a)+f(b)+4*\sum_{i=1}^{n-1} f(i)+2*\sum_{i=2}^{n-1} f(i)+2*\triangle x/3$$

Funktion eingeben

Die zu integrierende Funktion muß vor dem Starten des Programms in Zeile 250 eingegeben werden (LET F=...; z.B. LET F=2*x^2).

Eingabe der Werte

Nach dem Starten des Programms befinden Sie sich im Menü, wobei Sie zunächst "1" tippen müssen. Nun

werden die Integrationsgrenzen sowie die Toleranz, um die der errechnete Wert vom exakten Integral abweichen darf, verlangt. Anschließend erfolgt ein Rücksprung ins Menü.

Berechnung

Nach Aufruf dieses Unterprogramms stehen die Daten für die Ausgabe zur Verfügung.

Ausgabe

Damit geben Sie die zuletzt errechneten Werte wahlweise auf Drukker oder Bildschirm aus.

Wichtig: Da sämtliche trigonometrischen Funktionen im Bogenmaß berechnet werden, müssen zur Integration auch die Integrationsgrenzen im Bogenmaß angegeben sein.

Thomas Müller

```
  100 !Simpson-Integral
  110 !für Chip Special
  120 !von Thomas Müller
  130 !
  140 GOTO 280
  150 CALL MENUE
  160 !-- Ende --
  170 CLOSE #1
  180 CLOSE #2
  190 TEXT 40
  200 END
  210 !
  220 !**** Funktion ****
  230 !
  240 DEF F(X)
  250   LET F=X
  260 END DEF
  270 !
  280 !**** Initialisierung ****
  290 !
  300 CLEAR TEXT:SET STATUS OFF
  310 STRING H$(1)*16
  320 LET AS,UG,OG,FA,T,X,Y,US,ES,H
,FL,GS,FG=0
  330 LET A$=""
  340 LET H$(1)="-#.########^^^^"
  350 !--- window 1 ---
  360 SET VIDEO MODE 1
  370 SET VIDEO COLOUR 2
  380 SET VIDEO X 42
  390 SET VIDEO Y 5
  400 OPEN #1:"video:"
  410 CLEAR #1
  420 SET #1:PALETTE BLACK,BLUE,RED
,RGB(.5,.5,.6),0,0,0,0
  430 SET #1:INK 3
  440 PLOT #1:200,170,
  450 PRINT #1:"Simpson-Integral"
  460 PLOT #1:200,70;1200,70
  470 DISPLAY #1:AT 1 FROM 1 TO 5
  480 !--- window 2 ---
  490 SET VIDEO MODE 1
  500 SET VIDEO COLOUR 0
  510 SET VIDEO X 42
  520 SET VIDEO Y 5
  530 OPEN #2:"video:"
  540 CLEAR #2
  550 OPTION ANGLE DEGREES
  560 FOR I=0 TO 180 STEP 8
  570   LET H1=COS(I):LET H2=SIN(I)

  580   PLOT #2:300+H1*15,150+H2*20

  590   PLOT #2:270+H1*15,50-H2*20
  600 NEXT I
  610 PLOT #2:285,50;285,150
  620 PLOT #2:300,100,
  630 PRINT #2:"f(x) dx = ?";
  640 DISPLAY #2:AT 6 FROM 1 TO 5
  650 SET #102:PALETTE BLACK,BLUE,R
ED,0,0,0,0,0
  660 OPTION ANGLE RADIANS
  670 GOTO 150
  680 !
  690 !**** Menü ****
  700 !
  710 DEF MENUE
  720   CLEAR TEXT
  730   DISPLAY #102:AT 11 FROM 11
TO 24
  740   PRINT AT 15,10:"1 <= Eingab
e"
  750   PRINT AT 17,10:"2 <= Berech
nen des Integrals"
  760   PRINT AT 19,10:"3 <= Ausgab
e"
  770   PRINT AT 21,10:"4 <= Ende"
  780   CALL WAIT
  790   LET AS=VAL(A$)
  800   IF AS<1 OR AS>4 THEN 780
  810   PING
  820   PING
  830   SELECT CASE AS
  840   CASE 1
  850     !--- Grenzen ---
  860     CALL EINGABE
  870   CASE 2
  880     !--- Berechnen ---
```

```
  890      CALL RECHNEN
  900    CASE 3
  910      !--- Ausgabe ---
  920      CALL AUSGABE
  930    CASE 4
  940      !--- Ende ---
  950      GOTO 980
  960    END SELECT
  970    GOTO 720
  980  END DEF
  990  !
 1000  !**** Warteschleife ****
 1010  !
 1020  DEF WAIT
 1030    LET A$=INKEY$
 1040    IF A$="" THEN 1030
 1050  END DEF
 1060  !
 1070  !**** Eingabe der Daten ****
 1080  !
 1090  DEF EINGABE
 1100    CLEAR TEXT
 1110    PRINT AT 12,17:"Eingabe"
 1120    PRINT AT 13,17:"========"
 1130    INPUT AT 16,0,PROMPT "Unter
e Grenze: ":UG
 1140    INPUT AT 18,0,PROMPT "Obere
 Grenze:  ":OG
 1150    INPUT AT 20,0,PROMPT "Toler
anz:       ":T
 1160  END DEF
 1170  !
 1180  !**** Berechnung des Integral
s ****
 1190  !
 1200  DEF RECHNEN
 1210    LET FA=2:LET X=UG+(OG-UG)/2

 1220    LET US=F(X)
 1230    LET X=UG:LET Y=F(X):LET X=0
:LET ES=Y+F(X)
 1240    LET FA=FA*2:LET H=FL:LET FG
=(OG-UG)/FA
 1250    LET GS=GS+US:LET US=0
 1260    FOR I=1 TO FA/2
 1270      LET X=UG+FG*(2*I-1)
 1280      LET US=US+F(X)
 1290    NEXT I
 1300    LET FL=(ES+4*US+2*GS)*FG/3
 1310    IF ABS(FL-H)>ABS(T*FL) THEN
 1240
 1320  END DEF
 1330  !
 1340  !**** Ausgabe der Daten ****
 1350  !
 1360  DEF AUSGABE
 1370    CLEAR TEXT:LET DF=0
 1380    PRINT AT 14,10:"Ausgabe der
 Daten auf den"
 1390    INPUT AT 16,15,PROMPT "Druc
ker (j/n)? ":D$
 1400    IF D$="j" THEN LET DF=1:GOT
O 1420
 1410    IF D$<>"n" THEN 1390
 1420    CLEAR TEXT
 1430    PRINT AT 11,14:"Integraldat
en"
 1440    PRINT AT 12,14:"==========
=="
 1450    PRINT AT 15,5:"Untere Grenz
e:";
 1460    PRINT AT 15,20,USING H$(1):
UG
 1470    PRINT AT 17,5:"Obere Grenze
:";
 1480    PRINT AT 17,20,USING H$(1):
OG
 1490    PRINT AT 19,5:"Toleranz:";
 1500    PRINT AT 19,20,USING H$(1):
T
 1510    PRINT AT 21,5:"Feldanzahl:"
;
 1520    PRINT AT 21,20:FA
 1530    PRINT AT 23,5:"Fläche:"
 1540    PRINT AT 23,20,USING H$(1):
FL
 1550    PRINT AT 24,21:"_________
____";
 1560    IF DF=1 THEN COPY
 1570    CALL WAIT
 1580  END DEF
```


Suchen Sie ein ganz bestimmtes Programm für Ihre Problemlösung?

Wir bereiten laufend neue CHIP-Specials vor. Wenn Sie uns Ihr Problem und den Computer nennen, auf dem das Programm geschrieben werden soll, haben Sie die Chance, es in einem der nächsten CHIP-Specials zu finden. Über Ihren Vorschlag freut sich:

Die Redaktion CHIP-Special, Armin Schwarz

Mein Biorhythmus

Rechner: Enterprise 64/128
Programmname: Biorhythmus
Programmlänge: ca. 4700 Bytes
Programmiersprache: BASIC
Zubehör: Drucker optional

"Lebenslinien"

Gäbe es schon im alten Rom dieses Programm nebst zugehörigem Computer, Cäsar hätte es leichter gehabt. Denn so wäre er in der Lage gewesen, in den Iden des März seine Physis vorherzubestimmen.
In der Realität würde dies allerdings leider nicht funktionieren, da dieser Algorithmus "nur" für die Zeit vom 1.März 1900 bis zum 28.Februar 2100 gilt. Diese Einschränkung ist dadurch bedingt, daß diejenigen Jahre, die durch 100 teilbar sind, und bei welchen die ersten beiden Ziffern nicht durch vier teilbar sind (1700, 1800, 1900, etc.), keine Schaltjahre darstellen.
Ansonsten gilt die Regel, daß alle vier Jahre ein Schaltjahr eingefügt wird, das 366 Tage dauert (es gibt einen 29. Februar).

Entdeckung des Biorhythmus

Der Begründer dieser Theorie war ein gewisser Dr. Fließ, ein Berliner Hals- und Nasenchirurg. Dieser trat 1897 erstmals mit dem Gedanken an die Öffentlichkeit, daß das Leben in bestimmten Zyklen verlaufe. So war es für ihn Tatsache, daß zwei Grundperioden (23 und 28 Tage) unser gesamtes Leben, von der Geburt an, unterliegt. In späterer Zeit entdeckte man noch einen weiteren (33 Tage). Jedem Zyklus wird ein eigener Bereich des menschlichen Lebens zugeschrieben: So ist der 23-Tage-Zyklus für die körperliche, der 28-Tage-Zyklus für die seelische und die 33-Tage-Periode für die geistige Verfassung maßgeblich. Die kritischen Tage sind nach Meinung dieser Wissenschaftler die Schnittpunkte der Kurven mit der Mittellinie (0 %). Im übrigen können die einzelnen Gebiete bis zu -100 bzw. 100 Prozent erreichen.

Bedienung des Programms

Bei diesem Programm gibt es zwei verschiedene Ausgabemöglichkeiten: Zum einen auf den Bildschirm, wovon allerdings aufgrund der höchsten Auflösung (336*225 Punkte) keine Hardcopy auf den Drucker gegeben werden kann; zum anderen den direkten Ausdruck, der natürlich nicht so differnziert sein kann wie die Bildschirmdarstellung, da auf die hochauflösende Graphik zugunsten der Kompatibilität zu anderen Drukkern verzichtet wurde.
Doch zunächst zu den Eingaben:
Als erstes müssen Sie Ihr Geburtsdatum im Format TT-MM-JJJJ eingeben (das Trennzeichen ist beliebig). Nun wird nach dem Zeitraum gefragt, für den der Biorhythmus erstellt werden soll ("Prognose für MM-JJJJ"). Aus Gründen der Einfachheit kann nur der Monat und nicht das genaue Datum angegeben werden. (Beachte: Auch hier ist das Jahr vierstellig einzutippen!). Daraufhin erscheint die obig bereits erwähnte Abfrage, auf welches Gerät die Ausgabe erfolgen soll. Nach der Berechnung besteht die Möglichkeit für einen Neustart.
P.S. Sollten die Kurven Ihnen nichts Gutes für die Zukunft verheißen, so denken Sie daran: Es ist nur eine Theorie!

Thomas Müller

BEIM ABTIPPEN ARBEIT SPAREN
können Sie sich, wenn Sie mit der Anforderungskarte am Heftende die Programme auf Datenträger anfordern.
Anwenden und auf Ihre Belange abändern lassen sich diese Programme allerdings nur mit Hilfe dieser CHIP-SPECIAL-Ausgabe.

```
Biorhythmus
===========

Für den Monat Februar/ 1986
Geburtsdatum:  13 . 8 . 1966

--------------------------------------------------------------------------------
 1     :             x                     :               - o                 :
--------------------------------------------------------------------------------
 2     :                   x               :      -              o             :
--------------------------------------------------------------------------------
 3     :                          x        : -                         o       :
--------------------------------------------------------------------------------
 4     :                               -   x                              o    :
--------------------------------------------------------------------------------
 5     :                         -         :      x                       o    :
--------------------------------------------------------------------------------
 6     :                   -               :              x             o      :
--------------------------------------------------------------------------------
```

```
100 !Biorhythmus
110 !für Chip Special
120 !von Thomas Müller
130 !
140 !==== Initialisierung ====
150 !
160 CLEAR TEXT:SET STATUS OFF
170 DIM D(12),PE(3)
180 STRING WT$(7)*10,H$(1)*10,Z$(
3)*1,M$(12)*10,L$(1)*80,LI$(1)*80
190 LET H,H1,H2,H3,D3,M3,Y3=0
200 LET A$,H$(1),L$(1),LI$(1)=""
210 LET PE(1)=23:LET PE(2)=28:LET
PE(3)=33
220 LET Z$(1)="o":LET Z$(2)="x":L
ET Z$(3)="-"
230 FOR I=1 TO 12
240   READ M$(I)
250 NEXT I
260 FOR I=0 TO 12
270   READ D(I)
280 NEXT I
290 FOR I=1 TO 80
300   LET L$(1)=L$(1)&CHR$(32):LE
T LI$(1)=LI$(1)&CHR$(45)
310 NEXT I
320 !--- Daten für Monatsnamen --
-
330 DATA "Januar","Februar","März
","April"
340 DATA "Mai", "Juni","Juli","Au
gust"
350 DATA "September","Oktober"
360 DATA "November","Dezember"
370 !--- Daten für die Monate ---

380 DATA 0,31,59,90,120,151,181,2
12,243,273,304,334,365
390 GOTO 600
400 !--- Titelbild ---
410 SET VIDEO MODE 5
420 SET VIDEO COLOUR 1
430 SET VIDEO X 40
440 SET VIDEO Y 25
450 OPEN #1:"video:"
460 CLEAR #1
470 DISPLAY #1:AT 1 FROM 1 TO 25
480 PLOT #1:300,470,
490 PRINT #1:"Biorhythmus"
500 LET X=300:LET Y1=700:LET Y2=2
00
510 FOR I=0 TO 3*PI STEP PI/58
520   LET H=SIN(I)
530   PLOT #1:X,Y1+H*150
540   PLOT #1:X,Y2-H*150
550   LET X=X+4
560 NEXT I
570 PING
580 CALL WAIT
590 CLOSE #1
600 !--- window 2 ---
610 SET VIDEO MODE 1
620 SET VIDEO COLOUR 0
630 SET VIDEO X 42
640 SET VIDEO Y 25
650 OPEN #2:"video:"
660 SET #2:PALETTE BLACK,RGB(.7,.
2,.9),BLUE,RED,0,0,0,0
670 !--- Textfenster ---
680 SET #102:PALETTE BLACK,RGB(.6
,.6,.7),BLUE,RED,0,0,0,0
690 !
700 !==== Eingabe ====
710 !
720 CLEAR TEXT
730 DISPLAY #102:AT 1 FROM 1 TO 2
4
740 LET D(2)=59
```

```
  750 PRINT AT 3,15:"Eingabe"
  760 PRINT AT 4,15:"======="
  770 INPUT AT 8,5,PROMPT "Geburtsd
atum: ":H$(1)
  780 LET H=VAL(H$(1)(4:5))
  790 IF H<1 OR H>12 OR(IP(H)<>H) T
HEN 770
  800 LET M1=H
  810 LET H=VAL(H$(1)(1:2))
  820 IF H<1 OR H>(D(M1)-D(M1-1)) O
R(IP(H)<>H) THEN 770
  830 LET D1=H
  840 LET H=VAL(H$(1)(7:10))
  850 IF H<1 OR(IP(H)<>H) THEN 770
  860 LET Y1=H
  870 LET D3=D1:LET M3=M1:LET Y3=Y1
:CALL DAYS
  880 LET G1=H
  890 INPUT AT 10,5,PROMPT "Prognos
e für     ":H$(1)
  900 LET H=VAL(H$(1)(1:2))
  910 IF H<1 OR H>12 OR(IP(H)<>H) T
HEN 890
  920 LET M2=H
  930 LET H=VAL(H$(1)(4:7))
  940 IF H<1 OR(IP(H)<>H) THEN 890
  950 LET Y2=H
  960 LET D3=1:LET M3=M2:LET Y3=Y2:
CALL DAYS
  970 LET G2=H:LET G3=G2-G1
  980 IF IP(Y2/4)=Y2/4 AND M2>1 THE
N LET D(2)=60
  990 PRINT AT 12,5:"Bildschirm ode
r Drucker (B/D)?"
 1000 CALL WAIT
 1010 LET DF=0
 1020 IF A$="d" THEN LET DF=1:GOTO
1040
 1030 IF A$<>"b" THEN 1000
 1040 CLEAR TEXT
 1050 SELECT CASE DF
 1060   !--- Bildschirm ---
 1070 CASE 0
 1080   CLEAR #2
 1090   DISPLAY #2:AT 1 FROM 1 TO 2
5
 1100   LET X=0
 1110   PLOT #2:0,0;0,700
 1120   FOR I=1 TO D(M2)-D(M2-1)
 1130     PLOT #2:X,36,
 1140     PRINT #2:STR$(I);
 1150     LET X=X+43
 1160     PLOT #2:X-8,0;X-8,700
 1170   NEXT I
 1180   FOR I=1 TO D(M2)-D(M2-1)
 1190     FOR I2=1 TO 3
 1200       LET H1=G3+I-1:LET H2=I2
:CALL PERIOD
 1210       PLOT #2:I*43-40,H3*300+
400,
 1220       PRINT #2:Z$(I2);
 1230     NEXT I2
 1240   NEXT I
 1250   PLOT #2:0,400;(D(M2)-D(M2-1
))*43,400
 1260   PLOT #2:0,700;(D(M2)-D(M2-1
))*43,700
 1270   PLOT #2:0,60;(D(M2)-D(M2-1)
)*43,60
 1280   PLOT #2:300,899,
 1290   PRINT #2:"Monat: ";M$(M2);"
         Jahr:";Y2;
 1300   PLOT #2:100,800,
 1310   PRINT #2:"o = physisch
  x = seelisch          - = geistig"

 1320   CALL WAIT
 1330   !--- Drucker ---
 1340 CASE 1
 1350   LPRINT
 1360   PRINT #104:L$(1)(1:30);"Bio
rhythmus"
 1370   PRINT #104:L$(1)(1:30);"===
========"
 1380   LPRINT :LPRINT
 1390   LPRINT "Für den Monat ";M$(
M2);"/";Y2
 1400   LPRINT "Geburtsdatum: ";D1;
".";M1;".";Y1
 1410   LPRINT
 1420   LPRINT LI$(1)
 1430   FOR I=1 TO D(M2)-D(M2-1)
 1440     PRINT #104,USING "##":I;
 1450     PRINT #104:"     :";L$(1)
(1:35);":";L$(1)(1:35);":";CHR$(13)
;
 1460     FOR I2=1 TO 3
 1470       LET H1=G3+I-1:LET H2=I2
:CALL PERIOD
 1480       PRINT #104:L$(1)(1:43+H
3*35);Z$(I2);CHR$(13);
 1490     NEXT I2
 1500     LPRINT
 1510     LPRINT LI$(1)
 1520   NEXT I
 1530   LPRINT "      -100%";L$(1)(
1:64);"+100%"
 1540   LPRINT
 1550   LPRINT "o = physisch    x =
 seelisch     - = geistig"
 1560   LPRINT :LPRINT
 1570 CASE ELSE
 1580 END SELECT
 1590 PING
```

```
 1600 CLEAR TEXT
 1610 CLEAR #2
 1620 DISPLAY #102:AT 1 FROM 1 TO 2
4
 1630 PRINT AT 10,10:"Neubeginn (j/
n)?"
 1640 CALL WAIT
 1650 IF A$="j" THEN 720
 1660 IF A$<>"n" THEN 1640
 1670 CLOSE #2
 1680 TEXT 40
 1690 END
 1700 !
 1710 !==== Berechnung der Tage ===
=
 1720 !
 1730 DEF DAYS
 1740   LET H=0:LET AD=0
 1750   IF IP(Y3/4)=Y3/4 THEN
 1760     IF M3>2 THEN LET AD=1
 1770   END IF
 1780   LET H=AD+(Y3-1)*365+IP((Y3-
1)/4)
 1790   LET H=H+D(M3)+D3
 1800 END DEF
 1810 !
 1820 !==== Errechnen der Werte ===
=
 1830 !
 1840 DEF PERIOD
 1850   LET H3=H1/PE(H2)
 1860   LET H3=FP(H3)*PI*2
 1870   LET H3=SIN(H3)
 1880 END DEF
 1890 !
 1900 !==== Warteschleife ====
 1910 !
 1920 DEF WAIT
 1930   LET A$=INKEY$
 1940   IF A$="" THEN 1930
 1950 END DEF
```

Life is life

Programmlänge: 6027 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128/Drucker

LIFE ist ein Simulationsspiel, welches von John Horton Conway schon vor vielen Jahren entwickelt wurde. Ursprünglich als gruppentheoretisches Problem auf dem Papier durchgespielt, lief es später auf einem Großcomputer seiner Universität, wurde jedoch schon bald auf den aufkommenden Mikrocomputern mit Erfolg implementiert.
Der Grundgedanke dieses Lebensspiels ist die Simulation des Wachstums einer Zellkolonie. Natürlich erfolgt das Leben und Sterben der Zellen nicht willkürlich, sondern ähnlich wie in der Natur nach festen Regeln:
1. Jede Zelle mit 2 oder 3 Nachbarzellen überlebt.
2. Jede Zelle mit mehr als drei Nachbarn stirbt.
3. Jedes leere Flächenelement mit genau drei Nachbarzellen wird mit einer Zelle besetzt.
Bestimmt kommt jetzt beim Leser Kritik auf, da sich in der Natur das biologische Leben ganz gewiß nicht mit drei simplen Gesetzmäßigkeiten beschreiben läßt. Mit Sicherheit muß solchen Einwänden zugestimmt werden, doch sollte man dabei nicht außer acht lassen, daß die Wechselwirkung eines Lebewesen mit seiner Umwelt von einer beinahe unüberschaubar großen Zahl von Faktoren abhängt für die die Kapazität und Rechengeschwindigkeit eines Heimcomputers bei weitem nicht ausreicht. Auch ist es schwer, wenn nicht sogar unmöglich, alle Umweltfaktoren über einen Algorithmus präzis zu formulieren.
Doch soll LIFE die Realität auch nur nachbilden, eben simulieren, und nimmt für sich nicht in Anspruch, reale Lebensvorgänge exakt nachzuvollziehen. Schließlich wollte Conway nur einen Eindruck der Zusammenhänge der natürlichen Gesetzmäßigkeiten vermitteln.
Trotz allem macht es Spaß, die Entwicklung der Zellverbände zu beobachten. Aufblühende Zellkolonien, die sich aufblähen, um end-

lich zu zerplatzen. Andere wiederum, die in ihrer Entwicklung ein Gleichgewicht (statisch oder dynamisch) erreichen.
Das Programm ist durch den Einsatz der Menütechnik einfach zu bedienen, und durch Betätigen der einzelnen Menütasten zu starten. Der erste Menüpunkt legt nach Angabe der gewünschten Zellenzahl ein Feld an, welches die zufällig verteilten Zellen enthält. Wer seine Zellverbände selbst aufbauen möchte, kann dazu den Punkt '2' anwählen, welcher die Eingabe über Joystick ermöglicht. Unter Punkt '2' erfolgt das Setzen eines Punktes mit '#' und das Löschen mit der Leertaste. Verlassen wird die Routine mit 'Q' (Quit). Punkt '3' löscht den Bildschirm und die vorhandenen Zellen. Hardcopies der einzelnen Wachstumsphasen kann man mit '4' anfertigen. Diese Routine enthält die COPY-Funktion des Standard-IS-BASIC, die aber durch eine eigene Druck-Routine für beliebige Drucker ausgetauscht werden kann. Mit '5' schließlich startet man die Berechnung der einzelnen Zell-Generationen und mit '6' verläßt man LIFE und landet wieder im IS-BASIC-Interpreter.

Programmerläuterung:

Da LIFE sich streng an die Regeln des Strukturierten Programmierens hält, entfällt eine genaue Beschreibung der einzelnen Funktionen. Die Unterprogramme werden ausschließlich mit CALL oder über GOSUB angesprungen und über RETURN wieder verlassen. Die Probleme, die sich mit wilden Einsprüngen über GOTO ergeben wurden somit weitgehend vermieden. Auch dürften die ausführlichen REM-Zeilen die letzten Verständnisprobleme lösen helfen.

Zum Schluß noch einige Tips: Da der Rechen-Algorithmus in BASIC formuliert ist, kommt es zu einigen Wartezeiten. Nur sollte man dabei beachten, daß pro Generation genau 4232 Zellpositionen im 40-Zeichen-Textmodus berechnet werden. Abhilfe würde also nur eine in Assembler verfaßte Rechen-Routine schaffen. Auch könnte die hochauflösende Grafik mit ihren 344064 Bildschirmpunkten hervorragend zur Darstellung der Zellkolonien eingesetzt werden. Die zu überprüfenden 2752512 Positionen machen dann jedoch den Einsatz von Assembler wirklich unumgänglich.
Eine weitere interessante Möglichkeit ist der Ausbau von LIFE zum Strategiespiel. Die Grundidee ist der geschickte Einsatz der Conwayschen Regeln, um die gegnerische Kolonie durch gezieltes Setzen von Zellen schließlich zu vernichten oder nach einer vorgegebenen Zahl von Generationen zahlenmäßig zu überflügeln. Als Gegner kann ein Mensch oder (was eleganter, weil schwieriger) der Computer antreten.
Viel Spaß. Reiner Kunz

```
  100 PROGRAM "Game of Life"
  110 !
  120 NUMERIC FELD(25,25),DUMMY(25,
25)
  130 TEXT 40
  140 CALL ARRAYINIT
  150 CALL EROEFFNUNG
  160 CALL MENUE
  170 PRINT AT 18,27:"Menue ?"
  180 PING
  190 DO
  200   DO
  210     LET MENUE$=INKEY$
  220   LOOP UNTIL MENUE$<>""
  230 LOOP WHILE MENUE$<"1" OR MENU
E$>"7"
  240 PING
  250 PRINT AT 18,27:"        "
  260 ON VAL(MENUE$) GOSUB 1010,193
0,1130,2740,1310,280
  270 GOTO 170
  280 !
  290 ! ***********
  300 ! * Schluss *
  310 ! ***********
  320 !
  330 CLEAR SCREEN
  340 TEXT 80
```

```
  350 PRINT "Programm-Ende"
  360 END  ! Spielende
  370 !
  380 ! **************
  390 ! * Eröffnung *
  400 ! **************
  410 !
  420 DEF EROEFFNUNG
  430   CLEAR SCREEN
  440   PRINT AT 4,12:"Game of life
"
  450   PRINT AT 5,12:"============
"
  460   PRINT AT 8,3:"REGELN:"
  470   PRINT
  480   PRINT "  1. Jede Zelle mit
2 oder 3 Nachbarn       überlebt."
  490   PRINT
  500   PRINT "  2. Jede Zelle mit
mehr als 3 Nach-        barn stirbt
."
  510   PRINT
  520   PRINT "  3. Jedes leere Ele
ment mit genau          3 Nachbarn
erhält eine Zelle."
  530   !
  540   INPUT AT 21,11,PROMPT "Weit
er mit ENTER":DUMMY$
  550   CLEAR SCREEN
  560 END DEF
  570 !
  580 ! ************************
  590 ! * Random-Verteilung *
  600 ! ************************
  610 !
  620 DEF RANDOM
  630   RANDOMIZE
  640   FOR I=1 TO ZELLEN
  650     LET ZEILE=RND(25)+1
  660     LET SPALTE=RND(25)+1
  670     LET FELD(ZEILE,SPALTE)=1
  680   NEXT I
  690   CALL FELDAUSGABE
  700 END DEF
  710 !
  720 ! **********
  730 ! * Menue *
  740 ! **********
  750 !
  760 DEF MENUE
  770   PRINT AT 2,28:"LIFE"
  780   PRINT AT 5,27:"1:Computer"
  790   PRINT AT 7,27:"2:Joystick"
  800   PRINT AT 9,27:"3:Löschen"
  810   PRINT AT 11,27:"4:Drucken"
  820   PRINT AT 13,27:"5:Start"
  830   PRINT AT 15,27:"6:Ende"
  840 END DEF
  850 !
  860 ! ************************
  870 ! * Spielfeld-Ausgabe *
  880 ! ************************
  890 !
  900 DEF FELDAUSGABE
  910   FOR ZEILE=1 TO 24
  920     FOR SPALTE=1 TO 24
  930       IF FELD(ZEILE,SPALTE)=1
 THEN
  940         PRINT AT ZEILE,SPALTE
:CHR$(142);
  950       ELSE
  960         PRINT AT ZEILE,SPALTE
:CHR$(32);
  970       END IF
  980     NEXT SPALTE
  990   NEXT ZEILE
 1000 END DEF
 1010 !
 1020 ! *************************
 1030 ! * Eingabe der Zellen *
 1040 ! *************************
 1050 !
 1060 INPUT AT 18,27,PROMPT "Zellen
 ":ZELLEN
 1070 IF ZELLEN>0 AND ZELLEN<626 TH
EN GOTO 1100
 1080 PRINT AT 18,27:"             "

 1090 GOTO 1060
 1100 CALL RANDOM
 1110 PRINT AT 18,27:"             "

 1120 RETURN
 1130 !
 1140 ! *************************
 1150 ! * Löschen des Feldes *
 1160 ! *************************
 1170 !
 1180 FOR ZEILE=1 TO 25
 1190   FOR SPALTE=1 TO 25
 1200     LET FELD(ZEILE,SPALTE)=0
 1210   NEXT SPALTE
 1220 NEXT ZEILE
 1230 !
 1240 FOR ZEILE=1 TO 24
 1250   FOR SPALTE=1 TO 24
 1260     PRINT AT ZEILE,SPALTE:CHR
$(32);
 1270   NEXT SPALTE
 1280 NEXT ZEILE
 1290 !
 1300 RETURN
 1310 !
 1320 ! ****************************
```

```
1330 ! * Generationen berechnen *
1340 ! ****************************
1350 !
1360 PRINT AT 20,27:"Ich rechne !"

1370 FOR ZEILE=2 TO 23
1380   FOR SPALTE=2 TO 23
1390     IF FELD(ZEILE,SPALTE)=1 T
HEN GOTO 1500
1400     LET NACHBARN=0
1410     CALL NACHBARNSUMME
1420     !
1430     ! ################
1440     ! # 3. Bedingung #
1450     ! ################
1460     !
1470     IF NACHBARN<>3 THEN GOTO
1640
1480     LET DUMMY(ZEILE,SPALTE)=1

1490     GOTO 1650
1500     LET NACHBARN=-1
1510     CALL NACHBARNSUMME
1520     !
1530     ! ################
1540     ! # 1.Bedingung #
1550     ! ################
1560     !
1570     IF NACHBARN<2 OR NACHBARN
>3 THEN
1580       GOTO 1640
1590     ELSE
1600       LET DUMMY(ZEILE,SPALTE)
=1
1610     END IF
1620     !
1630     GOTO 1650
1640     LET DUMMY(ZEILE,SPALTE)=0

1650   NEXT SPALTE
1660 NEXT ZEILE
1670 PRINT AT 20,27:"          "

1680 PING
1690 !
1700 ! ****************************
***
1710 ! * Umspeichern der Generatio
n *
1720 ! ****************************
***
1730 !
1740 FOR ZEILE=1 TO 24
1750   FOR SPALTE=1 TO 24
1760     LET FELD(ZEILE,SPALTE)=DU
MMY(ZEILE,SPALTE)
1770   NEXT SPALTE
1780 NEXT ZEILE
1790 CALL FELDAUSGABE
1800 RETURN
1810 !
1820 ! ****************************
*****
1830 ! * Ermittlung der Nachbarnsu
mme *
1840 ! ****************************
*****
1850 !
1860 DEF NACHBARNSUMME
1870   FOR L=ZEILE-1 TO ZEILE+1
1880     FOR K=SPALTE-1 TO SPALTE+
1
1890       LET NACHBARN=NACHBARN+F
ELD(L,K)
1900     NEXT K
1910   NEXT L
1920 END DEF
1930 !
1940 ! ************************
1950 ! * Eingabe mit Joystick *
1960 ! ************************
1970 !
1980 LET ZEILE=1:LET SPALTE=1
1990 PRINT AT 20,27:"Zeile : ";ZEI
LE
2000 PRINT AT 22,27:"Spalte: ";SPA
LTE
2010 PRINT AT ZEILE,SPALTE:"";
2020 DO
2030   LET TASTATUR$=INKEY$
2040   CALL CURSORMOVE
2050 LOOP UNTIL TASTATUR$<>""
2060 !
2070 ! Cursor links
2080 !
2090 IF TASTATUR$=CHR$(184) THEN
2100   IF (SPALTE-1)<1 THEN
2110     GOTO 2020
2120   ELSE
2130     LET SPALTE=SPALTE-1
2140     GOTO 1990
2150   END IF
2160 END IF
2170 !
2180 ! Cursor rechts
2190 !
2200 IF TASTATUR$=CHR$(188) THEN
2210   IF (SPALTE+1)>24 THEN
2220     GOTO 2020
2230   ELSE
2240     LET SPALTE=SPALTE+1
2250     GOTO 1990
2260   END IF
2270 END IF
```

```
 2280 !
 2290 ! Cursor runter
 2300 !
 2310 IF TASTATUR$=CHR$(180) THEN
 2320   IF (ZEILE+1)>24 THEN
 2330     GOTO 2020
 2340   ELSE
 2350     LET ZEILE=ZEILE+1
 2360     GOTO 1990
 2370   END IF
 2380 END IF
 2390 !
 2400 ! Cursor rauf
 2410 !
 2420 IF TASTATUR$=CHR$(176) THEN
 2430   IF (ZEILE-1)<1 THEN
 2440     GOTO 2020
 2450   ELSE
 2460     LET ZEILE=ZEILE-1
 2470     GOTO 1990
 2480   END IF
 2490 END IF
 2500 !
 2510 ! Zelle löschen
 2520 !
 2530 IF TASTATUR$=CHR$(32) THEN
 2540   LET FELD(ZEILE,SPALTE)=0
 2550   PRINT AT ZEILE,SPALTE:CHR$(
32);
 2560   GOTO 1990
 2570 END IF
 2580 !
 2590 ! Zelle setzen
 2600 !
 2610 IF TASTATUR$="#" THEN
 2620   LET FELD(ZEILE,SPALTE)=1
 2630   PRINT AT ZEILE,SPALTE:CHR$(
142);
 2640   GOTO 1990
 2650 END IF
 2660 !
 2670 IF TASTATUR$="q" THEN
 2680   PRINT AT 20,27:"
"
 2690   PRINT AT 22,27:"
"
 2700   RETURN
 2710 END IF
 2720 !
 2730 GOTO 1990
 2740 !
 2750 ! ****************************
***
 2760 ! * Ausdrucken des Spielfelde
s *
 2770 ! ****************************
***
 2780 !
 2790 COPY
 2800 RETURN
 2810 !
 2820 ! ************************
 2830 ! * Arrays initialisieren *
 2840 ! ************************
 2850 !
 2860 DEF ARRAYINIT
 2870   FOR ZEILE=1 TO 25
 2880     FOR SPALTE=1 TO 25
 2890       LET FELD(ZEILE,SPALTE)=
0
 2900       LET DUMMY(ZEILE,SPALTE)
=0
 2910     NEXT SPALTE
 2920   NEXT ZEILE
 2930   !
 2940 END DEF
 2950 !
 2960 ! ********************
 2970 ! * Bewege den Cursor *
 2980 ! ********************
 2990 !
 3000 DEF CURSORMOVE
 3010   LET CURSOR=FELD(ZEILE,SPALT
E)
 3020   !
 3030   PRINT AT ZEILE,SPALTE:"*";
 3040   PRINT AT ZEILE,SPALTE:"O";
 3050   !
 3060   IF CURSOR=1 THEN
 3070     PRINT AT ZEILE,SPALTE:CHR
$(142);
 3080   ELSE
 3090     PRINT AT ZEILE,SPALTE:CHR
$(32);
 3100   END IF
 3110   !
 3120 END DEF
```

Bitte etwas Geduld!

Wenn sich Ihr Programm nach dem Abtippen nicht sofort meldet, so werfen Sie Ihren Rechner bitte nicht gleich an die Wand. Die Wartezeiten beim Starten eines Programmes betragen oft mehrere Sekunden. Das hat seinen guten Grund, denn der Enterprise überprüft zuvor, ob es schwerwiegende Strukturfehler im Programm gibt. Gleichzeitig setzt er Zeiger auf die einzelnen Unterprogramme, damit die Aufrufe der Routinen schnell vorgenommen werden können. Dieser Zeitverlust beim Programm-Start ist ein kleiner Preis für die Freiheiten, die das strukturierte Programmieren bietet.

Adressen

Rechner: Enterprise 64/128
Programmname: Adressen
Programmlänge: ca. 13 500 Bytes
Programmiersprache: BASIC
Zubehör: Drucker

Fahrig wischte sich Herr D., Manager eines Großkonzerns, über die schweißnasse Stirn: Seine Sekretärin hatte ihm soeben mitgeteilt, daß die gesuchte Geheimnummer seines Geschäftskollegen nicht aufzufinden war. Seufzend griff er zum Telefon, um den Konkursverwalter anzurufen. - Das weitere Schicksal von Herrn D. ist uns nicht bekannt. Was wir aber wissen, ist: Mit einer guten Adreßverwaltung wäre ihm das nicht passiert.
Damit Ihnen nicht dasselbe widerfährt, bieten wir Ihnen hier ein derartiges Programm an. So ist es in der Lage, bis zu 145 (auf dem Enterprise 128, Programm 0) Datensätze zu sieben Kriterien 25 Zeichen zu verwalten.
Nach dem Starten durch "START" oder "RUN" erscheint zunächst die Meldung: "Bitte warten - ich dimensioniere". Hierauf befinden Sie sich im Menü.

Das Menü

Hier können Sie durch Drücken des Cursor-Sticks nach oben bzw. unten das gewünschte Unterprogramm anwählen; mit <SPACE> rufen Sie es dann auf. Dabei stehen Ihnen folgende Möglichkeiten zur Verfügung:

Datei ergänzen

Mit dieser Routine kann man neue Datensätze eingeben. Nachdem die Daten einer Person eingegeben wurden (Anrede, Name, Straße, Wohnort, Telefonnr.), wird eine Besonderheit dieser Datenverwaltung klar: So ist es möglich, dem Datensatz eine bis zu 25 Zeichen lange Bemerkung zuzuordnen. Diese kann z.B. als Stichwort verwendet werden. Wurde auch diese Eingabe getätigt (alle Eingaben können auch durch Drücken von <ENTER> übergangen werden), so frägt das Pogramm "Alle Eingaben korrekt?"; ist dies nicht der Fall, so geben Sie "n" ein und die Daten werden noch einmal abgefragt. Haben Sie hingegen "j" gedrückt, erhalten Sie die Meldung "Weitere Eingaben?"; wollen Sie die Eingabe fortsetzen, so tippen Sie "j"; bei Verneinen wird der Datenstamm nach dem zuletzt von Ihnen gewählten Kriterium (Default ist "Name:") sortiert. Daraufhin wird ins Menü verzweigt.

Datensatz bearbeiten

Nun müssen Sie die Nummer des Datensatzes, den Sie bearbeiten wollen, eingeben. (Diese können Sie evtl. über eine Suchroutine abfragen). Da Sie aber im folgenden auch in der Datei "blättern" können, soll dies nur eine Erleichterung sein (damit Sie nicht immer vor- bzw. zurückblättern müssen).
Sobald der entsprechende Datensatz angezeigt wird, haben Sie folgende Alternativen:

* "+","-" (vorwärts,rückwärts)
  Damit lassen Sie sich den nächsten bzw. vorhergehenden Datensatz zeigen (sofern vorhanden, ansonsten wird der Befehl ignoriert).
* "d" (Drucken)
  Nun stehen Ihnen drei Druckformate zur Verfügung (Abb. 2); es wird jeweils nur der angezeigte Datensatz ausgedruckt.
* "ä" (Ändern)
  Dies gestattet Ihnen, die Daten zu ändern; falls Sie jedoch bei einem Kriterium nur "e" eingeben, so bleibt der Datensatz völlig unverändert. Auch hier erfolgt die Abfrage "Alle Eingaben korrekt?". Anschließend sortiert der Rechner die Datei nach dem von Ihnen zuletzt gewählten Kriterium.
* "l" (Löschen)
  Mit diesem Befehl sollten Sie

vorsichtig sein: Zwar muß diese Eingabe noch einmal bestätigt werden ("Sind Sie sicher?"); ist jedoch daraufhin der Datensatz gelöscht, besteht keine Möglichkeit, ihn zu "retten".Im Anschluß hieran sortiert der Computer wie bei "Ändern".

* "e" (Ende)
  Hiermit springen Sie ins Hauptmenü zurück.

Direktes Suchen

Bei dieser Suchroutine müssen Sie den genauen Suchbegriff eingeben: Selbstverständlich muß der Suchbegriff dem Kriterium, nach dem zuletzt sortiert wurde, zugeordnet sein. Achten Sie dabei auf Groß- bzw. Kleinschreibung!
Ist ein Datensatz mit diesem Element vorhanden, so wird dieser angezeigt; nun befinden Sie sich im "Daten bearbeiten"-Modus. Andernfalls gibt der Rechner "Kein entsprechender Datensatz vorhanden!" aus und verzweigt (nach Tastendruck) ins Menü. Auch diese Funktion kann mit "e" abgebrochen werden.

Joker-Suchen

Dieses Feature gestattet Ihnen, auch dann einen Datensatz zu suchen, wenn Sie den Suchbegriff nicht ganz genau wissen: Den "Rest" des Begriffs, den Sie nicht wissen, ersetzen Sie durch "*" (Beispiel: Falls Sie nicht wissen, ob die betreffende Person Maier oder Meyer heißt, geben Sie "M*" ein).
Falls ein Datensatz mit diesem Anfangskürzel gefunden wird, verzweigt der Computer in "Daten bearbeiten". Ansonsten verfährt der Rechner wie bei "Direktes Suchen".

Sortieren

In diesem Programmteil können Sie die Datei nach einem beliebigen Kriterium sortieren lassen; beachten Sie aber, daß bei der Eingabe des Kriteriums dieses genau wie bei den Ausgaben des Rechners geschrieben werden muß (Großbuchstabe, Doppelpunkt). Als Sortierverfahren wurde eine Variante des schnellen SHELL-Sort- Algorithmus' verwendet.

Abspeichern

Um die Daten (nicht das Programm!) abzuspeichern, müssen Sie nun eine Bezeichnung eingeben, unter der die Datei abgespeichert werden soll; tippen Sie "e" ein, so erfolgt ein Rücksprung ins Menü.

Einladen

Hier müssen Sie ebenfalls den Namen der Datei angeben, die eingeladen werden soll (zurück ins Menü mit "e"). Daraufhin werden die Daten eingelesen.

Funktionstasten

Unter diesem Menüpunkt können Sie eine Phrasensammlung anlegen: Den Funktionstasten kann je ein beliebiger Text zugewiesen werden (z.B. "Sehr geehrte Damen und Herren"); zusätzlich ist es möglich, durch "&" ein <ENTER> anzuhängen (z.B. "Sehr geehrter Herr&"; das Programm geht hierauf zum nächsten Kriterium über). Nach Programmende werden den Funktionstasten wieder die "normalen" Daten zugewiesen.

Wichtig: Benützen Sie im Eingabe-Modus nie den Cursorstick!!!

Thomas Müller

Druckmaske 1

```
Herrn
Peter Dussel
Faulenzerallee 34
57 Hinterdupfing 2
```

Druckmaske 2

```
Paula Fleißig
Fertigungsstraße 31
1111 Schlumpfhausen 2
Tel.: 234/765432461
```

Druckmaske 3

```
Muster, Alfred
Freizeitstraße 122a
6 Himmel 7
Tel.: 0034/1276/2244
Hobbybastler
```

```
100 !Adressen
110 !für Chip Special
120 !von Thomas Müller
130 !
140 GOTO 180
150 CALL MENUE
160 END
170 !
180 !**** init ****
190 !
200 CLEAR SCREEN:SET STATUS OFF
210 SET #102:PALETTE BLACK,RGB(.3
,.45,.5),RED,0,0,0,0,0
220 PRINT AT 12,5:"Bitte warten -
 ich dimensioniere"
230 PRINT AT 13,5:"--------------
------------------"
240 LET AN=50
250 STRING DA$(6,AN)*25,KR$(6)*12
,UP$(10)*30,S$(1)*25,AR$(3)*30,H$(6
)*25,C$(2)*25
260 DIM PO(AN+30),T(10),P(3)
270 FOR I=0 TO 10
280   READ UP$(I)
290   LET T(I)=22-IP(LEN(UP$(I))/
2)
300 NEXT I
310 FOR I=0 TO 6
320   READ KR$(I)
330 NEXT I
340 LET C$(1)="":LET C$(2)=""
350 FOR I=1 TO 25
360   LET C$(1)=C$(1)&CHR$(255)
370   LET C$(2)=C$(2)&CHR$(0)
380 NEXT I
390 FOR I=0 TO AN
400   LET PO(I+30)=I
410   FOR I2=0 TO 6
420     LET DA$(I2,I)=C$(1)
430   NEXT I2
440 NEXT I
450 LET A$=""
460 LET AS,AK,TH,EF,L,D,M,FF=0
470 LET K=1
480 LET P(0)=31:LET P(1)=15:LET P
(2)=7:LET P(3)=3
490 !--- Daten für UP$ ---
500 DATA "Datei ergänzen"
510 DATA "Daten bearbeiten"
520 DATA "Direktes Suchen"
530 DATA "Joker-Suchen"
540 DATA "Sortieren"
550 DATA "Abspeichern"
560 DATA "Einladen"
570 DATA "Funktionstasten"
580 DATA "Programm beenden"
590 DATA "Daten ändern"
600 DATA "Daten löschen"
610 !--- Daten für KR$ ---
620 DATA "Anrede:_____"
630 DATA "Name:_______"
640 DATA "Vorname:____"
650 DATA "Straße:_____"
660 DATA "Wohnort:____"
670 DATA "Telefon:____"
680 DATA "Bemerkung:__"
690 !--- windows eröffnen ---
700 SET VIDEO MODE 1
710 SET VIDEO COLOUR 1
720 SET VIDEO X 42
730 SET VIDEO Y 25
740 OPEN #1:"video:"
750 SET #1:PALETTE BLACK,BLUE,CYA
N,0,0,0,0,0
760 SET #1:CURSOR CHARACTER 32
770 SET #1:CHARACTER 140,192,224,
240,248,252,248,240,224,192
780 SET VIDEO MODE 5
790 SET VIDEO COLOUR 1
800 SET VIDEO X 42
810 SET VIDEO Y 4
820 OPEN #2:"video:"
830 SET #2:PALETTE BLACK,RGB(.8,.
8,.8),BLUE,BLUE,0,0,0,0
840 SET VIDEO MODE 2
850 SET VIDEO COLOUR 0
860 SET VIDEO X 42
870 SET VIDEO Y 1
880 OPEN #3:"video:"
890 SET #3:CHARACTER 64,60,60,60,
60,60,60,60,60,60
900 SET #3:PALETTE BLACK,BLUE,BLA
CK,0,0,0,0,0
910 GOTO 150
920 !
930 !**** Menü ****
940 !
950 DEF MENUE
960   DISPLAY #1:AT 1 FROM 1 TO 2
5
970   CLEAR #1
980   SET #1:INK 1
990   SET #1:LINE STYLE 7
1000  FOR I=0 TO 7
1010    PLOT #1:I*3,I*3;1343-I*3,
I*3
1020    PLOT #1:I*3,899-I*3;1343-
I*3,899-I*3
1030    PLOT #1:I*3,899-I*3;I*3,I
*3
1040    PLOT #1:1343-I*3,899-I*3;
1343-I*3,I*3
1050  NEXT I
1060  PLOT #1:600,840,
```

```
 1070    PRINT #1:"Menü"
 1080    PLOT #1:570,870;762,870;762
,762;570,762;570,870
 1090    SET #1:LINE STYLE 1
 1100    PLOT #1:602,802;730,802
 1110    PLOT #1:0,750,
 1120    PRINT #1:
 1130    FOR I=0 TO 8
 1140      PRINT #1:TAB(T(I));UP$(I)

 1150      PRINT #1:
 1160    NEXT I
 1170    !--- Auswahl ---
 1180    LET I=0:LET G=0
 1190    SET #1:INK 0
 1200    PLOT #1:T(G)*32-80,714-G*72
,
 1210    PRINT #1:CHR$(159);
 1220    SET #1:INK 2
 1230    PLOT #1:T(I)*32-80,714-I*72
,
 1240    PRINT #1:CHR$(140);
 1250    CALL WAIT
 1260    LET AS=ORD(A$)
 1270    IF AS=176 THEN LET G=I:LET
I=I-SGN(I):GOTO 1190
 1280    IF AS=180 THEN LET G=I:LET
I=I+SGN(8-I):GOTO 1190
 1290    IF AS<>32 THEN 1250
 1300    LET AS=I
 1310    SET #1:INK 0
 1320    PLOT #1:T(I)*32-80,714-I*72
,
 1330    PRINT #1:CHR$(159);
 1340    DISPLAY #3:AT 25 FROM 1 TO
1
 1350    CLEAR #3
 1360    !--- Aufrufe ---
 1370    SELECT CASE AS
 1380      !--- Datei ergänzen ---
 1390    CASE 0
 1400      CALL COMPLETE
 1410      CALL SORT_2
 1420      !--- Datensatz bearbeiten
---
 1430    CASE 1
 1440      CALL CHANGE_1
 1450      IF FF=1 THEN 1690
 1460      CALL CHANGE_2
 1470      !--- Direktes Suchen ---
 1480    CASE 2
 1490      CALL SEARCH_1
 1500      !--- Joker-Suchen ---
 1510    CASE 3
 1520      CALL SEARCH_2
 1530      !--- Sortieren ---
 1540    CASE 4
 1550      CALL SORT_1
 1560      !--- Abspeichern ---
 1570    CASE 5
 1580      CALL SAVE
 1590      !--- Einladen ---
 1600    CASE 6
 1610      CALL LOAD
 1620      !--- Funktionstasten ---
 1630    CASE 7
 1640      CALL FKEYS
 1650      !--- Programmende ---
 1660    CASE 8
 1670      CALL END
 1680    END SELECT
 1690    DISPLAY #1:AT 1 FROM 1 TO 2
5
 1700    GOTO 1170
 1710 END DEF
 1720 !
 1730 !**** Datei ergänzen ****
 1740 !
 1750 DEF COMPLETE
 1760    CLEAR TEXT
 1770    CALL SCREEN_1
 1780    CALL SCREEN_2
 1790    LET AK=AK+1
 1800    IF AK>AN THEN
 1810      LET AK=AK-1
 1820      PRINT AT 12,8:"Datei voll
!"
 1830      CALL WAIT
 1840      GOTO 3065
 1850    END IF
 1860    LET TH=TH+1:LET EF=0
 1870    FOR I=0 TO 6
 1880      PRINT AT 7+I*2,0:KR$(I);"
.........................."
 1890    NEXT I
 1900    FOR I=0 TO 6
 1910      INPUT AT 7+I*2,13,PROMPT
"":H$(1)
 1920      IF H$(1)="e"&".........
............" THEN
 1930        FOR I2=0 TO I
 1940          LET DA$(I2,AK)=C$(1)
 1950        NEXT I2
 1960        LET AK=AK-1:LET TH=TH-1

 1970        LET EF=1
 1980        EXIT FOR
 1990      END IF
 2000      LET DA$(I,AK)=H$(1):LET H
$(1)=""
 2010    NEXT I
 2020    IF EF=1 THEN 2180
 2030    PRINT AT 21,7:"Alle Eingabe
n korrekt (j/n)?"
```

```
 2040    CALL WAIT
 2050    IF A$="n" THEN
 2060      PRINT AT 21,36:"n"
 2070      GOTO 1900
 2080    END IF
 2090    IF A$<>"j" THEN 2040
 2100    PRINT AT 21,36:"j"
 2110    PRINT AT 23,10:"Weitere Ein
gaben (j/n)?"
 2120    CALL WAIT
 2130    IF A$="j" THEN
 2140      CLEAR TEXT
 2150      GOTO 1790
 2160    END IF
 2170    IF A$<>"n" THEN 2120
 2180    CLEAR TEXT
 2190  END DEF
 2200  !
 2210  !**** Daten bearbeiten ****
 2220  !
 2230  DEF CHANGE_1
 2240    LET FF=0
 2250    CALL SCREEN_1
 2260    CALL SCREEN_2
 2270    IF AK=0 THEN CALL EMPTY:LET
 FF=1:GOTO 2330
 2280    PRINT AT 12,10:"Welcher Dat
ensatz soll"
 2290    INPUT AT 14,10,PROMPT "bear
beitet werden? ":H$(1)
 2300    IF H$(1)="e" THEN LET FF=1:
GOTO 2330
 2310    LET D=VAL(H$(1))
 2320    IF D<1 OR D>AK THEN 2290
 2330  END DEF
 2340  !
 2350  !**** UP Daten bearbeiten ***
*
 2360  !
 2370  DEF CHANGE_2
 2380    CLEAR TEXT
 2390    DISPLAY #3:AT 25 FROM 1 TO
1
 2400    PRINT #3:"     § + vorwärts
 § - rückwärts § D Drucken § Ä Ände
rn § L Löschen § E Ende §";
 2410    CLEAR TEXT
 2420    PRINT AT 5,0:"Datensatz Nr.
:";D
 2430    FOR I=0 TO 6
 2440      PRINT AT 7+I*2,0:KR$(I);D
A$(I,PO(30+D))
 2450    NEXT I
 2460    CALL WAIT
 2470    SELECT CASE ORD(A$)
 2480      !--- vorwärts ---
 2490    CASE 43
 2500      LET D=D+SGN(AK-D)
 2510      GOTO 2410
 2520      !--- rückwärts ---
 2530    CASE 45
 2540      LET D=D-SGN(D-1)
 2550      GOTO 2410
 2560      !--- Drucken ---
 2570    CASE 100
 2580      CALL DRUCK
 2590      GOTO 2410
 2600      !--- Ändern ---
 2610    CASE 123
 2620      LET FF=0
 2630      FOR I=0 TO 6
 2640        INPUT AT 7+I*2,13,PROMP
T "":H$(I)
 2650        IF H$(I)="e"&".........
..............." THEN
 2660          LET FF=1
 2670          EXIT FOR
 2680        END IF
 2690      NEXT I
 2700      IF FF=1 THEN 2410
 2710      PRINT AT 21,5:"Alle Einga
ben korrekt (j/n)?"
 2720      CALL WAIT
 2730      IF A$="n" THEN
 2740        PRINT AT 21,34:"n"
 2750        GOTO 2630
 2760      END IF
 2770      IF A$<>"j" THEN 2720
 2780      FOR I=0 TO 6
 2790        LET DA$(I,PO(D+30))=H$(
I)
 2800      NEXT I
 2810      CALL SORT_2
 2820      GOTO 2410
 2830      !--- Löschen ---
 2840    CASE 108
 2850      PRINT AT 21,14:"Löschen:"

 2860      PRINT AT 23,8:"Sind Sie s
icher (j/n)?"
 2870      CALL WAIT
 2880      IF A$="n" THEN 2410
 2890      IF A$<>"j" THEN 2870
 2900      FOR I=0 TO 6
 2910        LET DA$(I,PO(30+D))=C$(
1)
 2920      NEXT I
 2930      LET TH=TH-1
 2940      CALL SORT_2
 2950      IF D>AK THEN LET D=AK
 2960      IF D<1 THEN LET D=1
 2970      IF AK=0 THEN 3050
 2980      GOTO 2410
 2990      !--- Ende ---
```

```
 3000    CASE 101
 3010      GOTO 3050
 3020    CASE ELSE
 3030    END SELECT
 3040    GOTO 2430
 3050 END DEF
 3060 !
 3070 !**** Direktes Suchen ****
 3080 !
 3090 DEF SEARCH_1
 3100    CLEAR TEXT
 3110    CALL SCREEN_1
 3120    CALL SCREEN_2
 3130    IF AK=0 THEN CALL EMPTY:GOT
O 3340
 3140    PRINT AT 9,13:"Suchbegriff:
"
 3150    PRINT AT 12,0:KR$(K);".....
....................."
 3160    INPUT AT 12,13,PROMPT "":S$
(1)
 3170    IF S$(1)(1:1)="e" THEN 3340

 3180    LET UG=0:LET OG=IP((AK+1)/2
)*2:LET FF=0
 3190    LET V=IP((UG+OG)/2)
 3200    IF S$(1)<DA$(K,PO(V+30)) TH
EN LET OG=V:GOTO 3230
 3210    IF S$(1)>DA$(K,PO(V+30)) TH
EN LET UG=V+1:GOTO 3230
 3220    GOTO 3310
 3230    IF UG>=OG THEN LET FF=FF+1
 3240    IF FF=2 THEN 3260
 3250    GOTO 3190
 3260    !--- nicht vorhanden ---
 3270    PRINT AT 16,6:"Kein entspre
chender Datensatz"
 3280    PRINT AT 18,15:"vorhanden!"

 3290    CALL WAIT
 3300    GOTO 3340
 3310    !--- gefunden ---
 3320    LET D=V
 3330    CALL CHANGE_2
 3340 END DEF
 3350 !
 3360 !**** Joker-Suchen ****
 3370 !
 3380 DEF SEARCH_2
 3390    CLEAR TEXT
 3400    CALL SCREEN_1
 3410    CALL SCREEN_2
 3420    IF AK=0 THEN CALL EMPTY:GOT
O 3610
 3430    PRINT AT 9,13:"Suchbegriff:
"
 3440    PRINT AT 12,0:KR$(K);".....
....................."
 3450    INPUT AT 12,13,PROMPT "":S$
(1)
 3460    IF S$(1)(1:1)="e" THEN 3610

 3470    LET H=POS(S$(1),"*")
 3480    LET S$(1)=S$(1)(1:H-1)&C$(2
)(1:26-H)
 3490    LET H=0
 3500    FOR I=1 TO AK
 3510      IF DA$(K,PO(30+I))>=S$(1)
THEN LET H=I:EXIT FOR
 3520    NEXT I
 3530    IF H=0 THEN
 3540      PRINT AT 16,6:"Kein entsp
rechender Datensatz"
 3550      PRINT AT 18,15:"vorhanden
!"
 3560      CALL WAIT
 3570      GOTO 3610
 3580    END IF
 3590    LET D=H
 3600    CALL CHANGE_2
 3610 END DEF
 3620 !
 3630 !**** Sortieren ****
 3640 !
 3650 DEF SORT_1
 3660    CLEAR TEXT
 3670    CALL SCREEN_1
 3680    CALL SCREEN_2
 3690    IF AK=0 THEN CALL EMPTY:GOT
O 3830
 3700    PRINT AT 9,12:"Sortierkrite
rium"
 3710    PRINT AT 12,14:"__________
_"
 3720    INPUT AT 12,14,PROMPT "":H$
(1)
 3730    IF H$(1)(1:1)="e" THEN 3830

 3740    LET FF=1
 3750    FOR I=0 TO 6
 3760      IF H$(1)=KR$(I) THEN
 3770        LET K=I:LET FF=0
 3780        EXIT FOR
 3790      END IF
 3800    NEXT I
 3810    IF FF=1 THEN 3710
 3820    CALL SORT_2
 3830 END DEF
 3840 !
 3850 !**** Abspeichern ****
 3860 !
 3870 DEF SAVE
 3880    CALL SCREEN_1
 3890    CALL SCREEN_2
```

```
 3900    PRINT AT 10,10:"Dateiname:
    "
 3910    INPUT AT 10,21,PROMPT "":H$
(1)
 3920    IF H$(1)(1:1)="e" THEN 4030
 3930    LET H$(1)=H$(1)(1:4)&"_adr"
 3940    OPEN #8:H$(1) ACCESS OUTPUT
 3950    PRINT #8:AK
 3960    PRINT #8:K
 3970    FOR I=1 TO AK
 3980      FOR I2=0 TO 6
 3990        PRINT #8:DA$(I2,PO(30+I
))
 4000      NEXT I2
 4010    NEXT I
 4020    CLOSE #8
 4030  END DEF
 4040  !
 4050  !**** Einladen ****
 4060  !
 4070  DEF LOAD
 4080    CALL SCREEN_1
 4090    CALL SCREEN_2
 4100    PRINT AT 10,10:"Dateiname:
    "
 4110    INPUT AT 10,21,PROMPT "":H$
(1)
 4120    IF H$(1)(1:4)="e___" THEN 4
270
 4130    LET H$(1)=H$(1)(1:4)&"_adr"
 4140    OPEN #8:H$(1) ACCESS INPUT
 4150    INPUT #8:AK
 4160    INPUT #8:K
 4170    FOR I=1 TO AK
 4180      FOR I2=0 TO 6
 4190        INPUT #8:DA$(I2,I)
 4200      NEXT I2
 4210    NEXT I
 4220    CLOSE #8
 4230    LET TH=AK
 4240    FOR I=0 TO AK
 4250      LET PO(I+30)=I
 4260    NEXT I
 4270  END DEF
 4280  !
 4290  !**** Funktionstasten ****
 4300  !
 4310  DEF FKEYS
 4320    CLEAR TEXT
 4330    CALL SCREEN_1
 4340    CALL SCREEN_2
 4350    FOR I=1 TO 16
 4360      CLEAR TEXT
 4370      PRINT AT 10,0:"Funktionst
aste ";I;":"
 4380      INPUT AT 12,0,PROMPT "":H
$(1)
 4390      IF H$(1)(1:1)="e" THEN 44
30
 4395      IF H$(1)="" THEN 4420
 4400      IF H$(1)(LEN(H$(1)):LEN(H
$(1)))="&" THEN LET H$(1)=H$(1)(1:L
EN(H$(1))-1)&CHR$(13)
 4410      SET #102:FKEY I H$(1)
 4420    NEXT I
 4430  END DEF
 4440  !
 4450  !**** Shell-Sort ****
 4460  !
 4470  DEF SORT_2
 4480    CLEAR TEXT
 4490    PRINT AT 10,10:"Datenstamm
wird nach dem"
 4500    LET H=POS(KR$(K),"_")
 4510    PRINT AT 12,9:"Kriterium ";
CHR$(34);KR$(K)(1:H-2);CHR$(34);" s
ortiert."
 4520    FOR M=0 TO 3
 4530      CALL SORT_UP
 4540    NEXT M
 4550    !--- Last Sort ---
 4560    FOR I=32 TO AK+30
 4570      LET PO(30)=PO(I):LET L=I-
1
 4580      IF DA$(K,PO(L))>DA$(K,PO(
30)) THEN
 4590        LET PO(L+1)=PO(L):LET L
=L-1:GOTO 4580
 4600      END IF
 4610      LET PO(L+1)=PO(30)
 4620    NEXT I
 4630    LET AK=TH
 4640    CLEAR TEXT
 4650  END DEF
 4660  !--- Sort-Unterprg ---
 4670  DEF SORT_UP
 4680    IF 2*P(M)>AK THEN 4770
 4690    FOR I=31 TO P(M)+30
 4700      FOR I2=I+P(M) TO AK+30-P(
M) STEP P(M)
 4710        LET PO(I-P(M))=PO(I2):L
ET L=I2-P(M)
 4720        IF DA$(K,PO(L))<=DA$(K,
PO(I-P(M))) THEN 4740
 4730        LET PO(L+P(M))=PO(L):LE
T L=L-P(M):GOTO 4720
 4740        LET PO(L+P(M))=PO(I-P(M
))
 4750      NEXT I2
 4760    NEXT I
 4770  END DEF
 4780  !
```

```
 4790 !**** Tastendruck ****
 4800 !
 4810 DEF WAIT
 4820   LET A$=INKEY$
 4830   IF A$="" THEN 4820
 4840 END DEF
 4850 !
 4860 !**** oberer Bildschirm ****
 4870 !
 4880 DEF SCREEN_1
 4890   CLEAR #2
 4900   DISPLAY #2:AT 1 FROM 1 TO 4

 4910   SET #2:PALETTE BLACK,RGB(.8
,.8,.8),BLUE,BLUE,0,0,0,0
 4920   SET #2:INK 1
 4930   PLOT #2:T(AS)*16,72,
 4940   PRINT #2:UP$(AS)
 4950   PLOT #2:T(AS)*16,20;(LEN(UP
$(AS))*64+T(AS)*16),20
 4960 END DEF
 4970 !
 4980 !**** Text-Bildschirm ****
 4990 !
 5000 DEF SCREEN_2
 5010   CLEAR TEXT
 5020   DISPLAY #102:AT 5 FROM 5 TO
 24
 5030 END DEF
 5040 !
 5050 !**** Datei leer ****
 5060 !
 5070 DEF EMPTY
 5080   PRINT AT 10,14:"Datei leer:
"
 5090   PRINT AT 12,10:"Keine Daten
 verfügbar!"
 5100   CALL WAIT
 5110 END DEF
 5120 !
 5130 !**** Programmende ****
 5140 !
 5150 DEF END
 5160   SET #102:FKEY 1 "START"&CHR
$(13)
 5170   SET #102:FKEY 2 "LIST"&CHR$
(13)
 5180   SET #102:FKEY 3 "AUTO"&CHR$
(13)
 5190   SET #102:FKEY 4 "TOGGLE REM
1"&CHR$(13)
 5200   SET #102:FKEY 5 "TEXT"&CHR$
(13)
 5210   SET #102:FKEY 6 "GRAPHICS"&
CHR$(13)
 5220   SET #102:FKEY 7 "TOGGLE KEY
 CLICK"&CHR$(13)
 5230   SET #102:FKEY 8 "INFO"&CHR$
(13)
 5240   SET #102:FKEY 9 "CONTINUE"&
CHR$(13)
 5250   SET #102:FKEY 10 "LLIST"&CH
R$(13)
 5260   SET #102:FKEY 11 "RENUMBER"
&CHR$(13)
 5270   SET #102:FKEY 12 "TOGGLE RE
M2"&CHR$(13)
 5280   SET #102:FKEY 13 "DISPLAY T
EXT"&CHR$(13)
 5290   SET #102:FKEY 14 "DISPLAY G
RAPHICS"&CHR$(13)
 5300   SET #102:FKEY 15 "TOGGLE SP
EAKER"&CHR$(13)
 5310   SET #102:FKEY 16 "TYPE"&CHR
$(13)
 5320   CLOSE #1
 5330   TEXT 40
 5340   END
 5350 END DEF
 5360 !
 5370 !**** Zurückstellen der Point
er ****
 5380 !
 5390 DEF HELP
 5400   FOR I=0 TO AN
 5410     LET PO(I+30)=I
 5420   NEXT I
 5430 END DEF
 5440 !
 5450 !**** Ausdruck ****
 5460 !
 5470 DEF DRUCK
 5480   PRINT AT 21,5:"Welches Druc
kformat (1-3)? "
 5490   CALL WAIT
 5500   IF A$="e" THEN 5800
 5510   LET H=ORD(A$)-48
 5520   IF H<1 OR H>3 THEN 5490
 5530   SELECT CASE H
 5540   CASE 1
 5550     LPRINT :LPRINT
 5560     LPRINT DA$(0,PO(30+D))(1:
POS(DA$(0,PO(30+D)),".")-1)
 5570     LPRINT DA$(2,PO(30+D))(1:
POS(DA$(2,PO(30+D)),".")-1);" ";
 5580     LPRINT DA$(1,PO(30+D))(1:
POS(DA$(1,PO(30+D)),".")-1)
 5590     LPRINT DA$(3,PO(30+D))(1:
POS(DA$(3,PO(30+D)),".")-1)
 5600     LPRINT DA$(4,PO(30+D))(1:
POS(DA$(4,PO(30+D)),".")-1)
 5610     LPRINT
 5620   CASE 2
 5630     LPRINT :LPRINT
```

```
 5640      LPRINT DA$(2,PO(30+D))(1:
POS(DA$(2,PO(30+D)),".")-1);" ";
 5650      LPRINT DA$(1,PO(30+D))(1:
POS(DA$(1,PO(30+D)),".")-1)
 5660      LPRINT DA$(3,PO(30+D))(1:
POS(DA$(3,PO(30+D)),".")-1)
 5670      LPRINT DA$(4,PO(30+D))(1:
POS(DA$(4,PO(30+D)),".")-1)
 5680      LPRINT "Tel.: ";
 5690      LPRINT DA$(5,PO(30+D))(1:
POS(DA$(5,PO(30+D)),".")-1)
 5700    CASE 3
 5710      LPRINT :LPRINT
 5720      LPRINT DA$(1,PO(30+D))(1:
POS(DA$(1,PO(30+D)),".")-1);", ";
 5730      LPRINT DA$(2,PO(30+D))(1:
POS(DA$(2,PO(30+D)),".")-1)
 5740      LPRINT DA$(3,PO(30+D))(1:
POS(DA$(3,PO(30+D)),".")-1)
 5750      LPRINT DA$(4,PO(30+D))(1:
POS(DA$(4,PO(30+D)),".")-1)
 5760      LPRINT "Tel.: ";
 5770      LPRINT DA$(5,PO(30+D))(1:
POS(DA$(5,PO(30+D)),".")-1)
 5780      LPRINT DA$(6,PO(30+D))(1:
POS(DA$(6,PO(30+D)),".")-1)
 5790    END SELECT
 5800 END DEF
```

Universal-Datei

Programmlänge: 21800 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128/Floppy/Drucker

Datenbanken sind eines der Haupteinsatzgebiete von Computern. Selbstverständlich, daß wir in diesem Special auch ein Datenbankprogramm veröffentlichen.
Dabei haben wir den Programmaufbau derart universell gestaltet, daß sich 'UNIDAT' praktisch wie ein Chamäleon an die verschiedensten Anforderungen anpassen läßt, ohne etwa große Programmierkenntnisse zu besitzen. Es genügt schon, die Menü-Texte entsprechen zu modifizieren.
Als Beispiel haben wir mit dem vorliegenden Programm eine Schallplatten-Datei realisiert. Schallplatten sind schließlich ein schönes Hobby, doch wie soll man einzelne LP's oder gar Songs in einem Wust von mehereren hundert Platten wiederfinden ? Kein Problem für Ihren Enterprise und dem Programm 'UNIDAT'.
Es werden dabei nicht nur Singles, sondern auch komplette Langspiel-Platten und sogar Samplers (also LP's mit mehreren Interpreten) verwaltet. Es genügt die einfache Eingabe des gesuchten Begriffes. Die Speicherung der Daten erfolgt in einer sequentiellen Datei, die vor dem ersten Programmlauf von Hand initialisiert werden muß. Dazu gibt man die folgenden BASIC-Befehle ein, deren vorgegebene Reihenfolge unbedingt eingehalten werden muß.
Zuerst schiebt man eine formatierte Datendiskette in das Laufwerk. Dann gibt man folgende Befehlsfolge über die Tastatur des Computers ein und generiert damit einen Kopfeintrag in die Datei:

```
OPEN#106:"LPDATEI.DAT"ACCESS OUTPUT
PRINT#106:"1"
PRINT#106:"01.01.0001"
PRINT#106:"0001"
PRINT#106:"SÄNGER"
PRINT#106:"TITEL"
PRINT#106:"1"
PRINT#106:"SONG1"
CLOSE#106
```

Danach kann das eigentliche Hauptprogramm mit 'RUN' gestartet werden. Nun erscheint das Hauptmenü mit den drei auszuwählenden Funktionen.
Taste '1' erlaubt die Erstellung und Bearbeitung eines Datensatzes. Über diese Funktion werden die einzelnen Einträge in die sequentielle Datei vorgenommen. Nach der Aktivierung erscheint ein Hilfsmenü, in welchem der Benutzer dazu aufgefordert wird, die gewünschte Datendiskette einzulegen und das

aktuelle Datum einzutippen. Darauf beginnt das Programm, die Datei nach dem letzten Eintrag abzusuchen und die laufende Nummer auf dem Bildschirm auszugeben. Der Benutzer wird nun aufgefordert, das Erscheinungsjahr und den Namen des Interpreten einzugeben. Anschließend wird der LP-Titel erfragt und die Songs der genannten Schallplatte eingegeben.
'Samplers' erfasst man durch Eingabe von 'DIV' bei der Abfrage des Interpreten. Die folgenden Schritte bleiben die selben.
Die Eingabe beendet man durch Eingabe von 'E', worauf die neuen Daten auf Diskette abgespeichert werden. Fehleingaben können mit 'F' korrigiert werden (das falsche Datum wird einfach neu eingegeben). Möchte man noch weitere Einträge vornehmen, gibt man auf die Frage des Programms ('Weiter (J/N)?') ein 'J' ein.
Taste '2' ermöglicht dem Benutzer das Suchen im Datenbestand. Dazu wird die gewünschte Datendiskette in das Laufwerk eingelegt und die Druckeroption auf Wunsch aktiviert. Dabei werden alle Ausgaben des Programms auf dem Drucker mitprotokolliert.
Die Suche erfolgt nun nach maximal sechs verschiedenen Kriterien (siehe auch Bildschirmausdruck).
Das Druckerprotokoll kann durch Taste '9' beendet werden. Ebenfalls kann über das Hilfsmenü mit Taste '7' eine neue Datendiskette eingelegt und das Druckerprotokoll erneut aktiviert werden.
Taste '3' schliesst alle offenen Datenkanäle und beendet das Programm.

Bedienungshinweise:

Bei der Eingabe von Musikgruppen sollte das 'The' fortgelassen werden, um die Suche mit Stichwörtern zu erleichtern.
Beachtenswert ist auch der Programmtrick, mit dem an bereits bestehende sequentielle Dateien neue Daten angehängt werden.
Das ist deshalb interessant, weil der Enterprise keine 'APPEND'-Möglichkeit für sequentielle Dateien kennt. Stattdessen muß die alte Datei mit dem Namen 'LPDATEI.DAT' in die Zwischendatei mit dem Namen 'LPDATEI.BAK' umbenannt werden. Nun kann diese Datei wieder nach 'LPDATEI.DAT' zurückkopiert und anschließend die neuen Daten angehängt werden.

Reiner Kunz

```
100 PROGRAM "UNIDAT"
110 !
120 TEXT 40
130 LET DR=0
140 LET NUMMER=0
150 LET W=0
160 LET N=0
170 LET S=0
180 LET H=0
190 LET I=0
200 STRING DIV$(100)*20
210 STRING SONG$(100)*30
220 LET LEZTESDATUM$=""
230 LET EINGABEDATUM$=""
240 !
250 ! ********************
260 ! * MENUE ANZEIGEN *
270 ! ********************
280 !
290 CLEAR SCREEN
300 !
310 PRINT AT 3,7:"SCHALLPLATTEN"
320 PRINT AT 4,7:"    DATEI    "
330 PRINT AT 5,7:"=============="
340 PRINT AT 11,1:"Bearbeitung de
s Datenbestandes.......1"
350 PRINT AT 14,1:"Suche Interpre
ten, LP-Titel, Songs...2"
360 PRINT AT 17,1:"Beenden des Pr
ogrammes..............3"
370 PRINT AT 22,2:"Bitte Eingabe
für Menü: ";
380 !
390 INPUT A$
400 IF A$="" THEN 390
410 LET A=VAL(A$)
420 ON A GOTO 570,2150,460
430 PING
440 GOTO 250
450 !
460 ! ************
470 ! * PRG-ENDE *
```

```
  480 ! ************
  490 !
  500 CLEAR SCREEN
  510 PRINT AT 11,15:"Ende der"
  520 PRINT AT 12,10:"Schallplatten
 - Datei"
  530 !
  540 TEXT 80
  550 END
  560 !
  570 ! ********************
  580 ! * FILE-VERWALTUNG *
  590 ! ********************
  600 !
  610 CLEAR SCREEN
  620 !
  630 PRINT AT 2,7:"EINGABE-MODUS"
  640 PRINT AT 4,7:"-------------"
  650 PRINT AT 7,2:"Sie befinden si
ch jetzt im Eingabe-"
  660 PRINT AT 8,2:"Modus des Progr
ammes ."
  670 PRINT AT 9,2:"Er erfasst Einz
el-LP's und sogenannte"
  680 PRINT AT 10,2:"Samplers ."
  690 PRINT AT 11,2:"Diese Samplers
 sind dadurch kenntlich"
  700 PRINT AT 12,2:"zu machen, daß
 man bei der Frage"
  710 PRINT AT 13,2:"nach dem INTER
PRETEN ein DIV eingibt."
  720 PRINT AT 16,2:"Datum >TT.MM.J
JJJ<: ";
  730 LINE INPUT EINGABEDATUM$
  740 PRINT AT 18,2:"Befindet sich
die Daten-Diskette"
  750 PRINT AT 19,2:"im Laufwerk 1
(J) ? ";
  760 !
  770 LET A$=INKEY$
  780 IF A$="" THEN 770
  790 IF A$="J" OR A$="j" THEN PRIN
T "Ja";:GOTO 830
  800 PING
  810 GOTO 740
  820 !
  830 PRINT AT 22,2:"Die Datei wird
 gelesen. Bitte WARTEN."
  840 GOSUB 1860
  850 !
  860 ! ****************
  870 ! * DATENEINGABE *
  880 ! ****************
  890 !
  900 CLEAR SCREEN
  910 !
  920 PRINT AT 1,2:"Bitte die Daten
 eingeben."
  930 PRINT AT 2,2:"Fehler=>F<  End
e=>E<  Sampler=>DIV<"
  940 !
  950 !
  960 LET NUMMER=NUMMER+1:LET W=14:
LET N=1:LET S=1
  970 PRINT AT 4,2:;:CALL SPC(37):P
RINT AT 4,2:"Nummer der LP    : ";S
TR$(NUMMER)
  980 PRINT AT 6,2:;:CALL SPC(37):P
RINT AT 6,2:"Letzte Eingabe am: ";L
EZTESDATUM$
  990 PRINT AT 8,2:"Eingabe-Datum
   : ";EINGABEDATUM$
 1000 PRINT AT 10,2:;:CALL SPC(37):
LINE INPUT AT 10,2,PROMPT "Erschein
ungsjahr : ":DATUM$
 1010 PRINT AT 11,2:;:CALL SPC(37):
LINE INPUT AT 11,2,PROMPT "Interpre
ten      : ":SAENGER$
 1020 IF RTRIM$(SAENGER$)="F" THEN
LET DATUM$="":GOTO 1000
 1030 PRINT AT 12,2:;:CALL SPC(37):
LINE INPUT AT 12,2,PROMPT "Titel de
r LP     : ":TITEL$
 1040 IF RTRIM$(TITEL$)="F" THEN LE
T SAENGER$="":GOTO 1010
 1050 PRINT AT 14,2:;:CALL SPC(37):
LINE INPUT AT 14,2,PROMPT "Song
   : ":SONG$(S)
 1060 IF RTRIM$(SONG$(S))="F" THEN
LET TITEL$="":GOTO 1030
 1070 IF RTRIM$(SAENGER$)="DIV" THE
N 1420
 1080 LET S=S+1:LET W=W+1
 1090 IF W>=23 THEN LET W=14
 1100 PRINT AT W,2:;:CALL SPC(37):L
INE INPUT AT W,2,PROMPT "Song
: ":SONG$(S)
 1110 IF RTRIM$(SONG$(S))="F" THEN
1140
 1120 IF RTRIM$(SONG$(S))="E" THEN
1150
 1130 GOTO 1080
 1140 LET SONG$(S)="":LET S=S-1:LET
 W=W-1:GOTO 1100
 1150 LET SONG$(S)="":LET S=S-1
 1160 !
 1170 ! ************************
 1180 ! * AUSGABE AUF DISKETTE *
 1190 ! ************************
 1200 !
 1210 GOSUB 7980 ! DATEI UMBENENNEN

 1220 OPEN #106:"disk:LPDATEI.DAT"
ACCESS OUTPUT
```

```
 1230 GOSUB 8100 ! DATEI UMBENENNEN
 1240 PRINT #106:STR$(NUMMER)
 1250 PRINT #106:EINGABEDATUM$
 1260 PRINT #106:DATUM$
 1270 PRINT #106:SAENGER$
 1280 PRINT #106:TITEL$
 1290 PRINT #106:STR$(S)
 1300 FOR I=1 TO S
 1310   PRINT #106:SONG$(I)
 1320 NEXT I
 1330 CLOSE #106
 1340 PRINT AT 23,1:"Noch mehr Date
n eingeben (J/N) ? ";
 1350 LET A$=INKEY$
 1360 IF A$="" THEN 1350
 1370 IF A$="J" OR A$="j" THEN PRIN
T AT 23,1:;:CALL SPC(38):GOTO 960
 1380 IF A$="N" OR A$="n" THEN 250
 1390 PING
 1400 GOTO 1340
 1410 !
 1420 ! ************************
 1430 ! * DATENEINGABE SAMPLER *
 1440 ! ************************
 1450 !
 1460 LET W=W+1
 1470 PRINT AT W,2:;:CALL SPC(37):L
INE INPUT AT W,2,PROMPT "Interpret
: ":DIV$(S)
 1480 IF RTRIM$(DIV$)="F" THEN LET
SONG$="":GOTO 1050
 1490 LET S=S+1:LET W=W+1
 1500 IF W>=23 THEN LET W=14
 1510 PRINT AT W,2:;:CALL SPC(37):L
INE INPUT AT W,2,PROMPT "Song
: ":SONG$(S)
 1520 IF RTRIM$(SONG$(S))="E" THEN
1580
 1530 IF RTRIM$(SONG$(S))="F" THEN
1570
 1540 LET W=W+1
 1550 PRINT AT W,2:;:CALL SPC(37):L
INE INPUT AT W,2,PROMPT "Interpret
: ":DIV$(S)
 1560 GOTO 1490
 1570 LET SONG$(S)="":LET DIV$(S)="
":LET W=W-2:LET S=S-1:GOTO 1510
 1580 LET SONG$(S)="":LET S=S-1:GOT
O 1600
 1590 !
 1600 ! ************************
 1610 ! * AUSGABE AUF DISKETTE *
 1620 ! ************************
 1630 !
 1640 GOSUB 7980 ! DATEI UMBENENNEN
 1650 OPEN #106:"disk:LPDATEI.DAT"
ACCESS OUTPUT
 1660 GOSUB 8100 ! DATEI UMKOPIEREN
 1670 PRINT #106:STR$(NUMMER)
 1680 PRINT #106:EINGABEDATUM$
 1690 PRINT #106:DATUM$
 1700 PRINT #106:SAENGER$
 1710 PRINT #106:TITEL$
 1720 PRINT #106:STR$(S)
 1730 FOR I=1 TO S
 1740   PRINT #106:SONG$(I)
 1750   PRINT #106:DIV$(I)
 1760 NEXT I
 1770 CLOSE #106
 1780 PRINT AT 23,2:"Noch mehr Date
n eingeben (J/N) ?";
 1790 LET A$=INKEY$
 1800 IF A$="" THEN 1790
 1810 IF A$="J" OR A$="j" THEN PRIN
T AT 23,2:;:CALL SPC(37):GOTO 960
 1820 IF A$="N" OR A$="n" THEN 250
 1830 PING
 1840 GOTO 1780
 1850 !
 1860 ! ******************************
 1870 ! * DATEN-DISKETTE EINLESEN *
 1880 ! ******************************
 1890 !
 1900 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
 1910 !
 1920 LINE INPUT #106,IF MISSING 20
50:NUMMER$
 1930 LET NUMMER=VAL(NUMMER$)
 1940 LINE INPUT #106:LEZTESDATUM$
 1950 LINE INPUT #106:DATUM$
 1960 LINE INPUT #106:SAENGER$
 1970 LINE INPUT #106:TITEL$
 1980 LINE INPUT #106:S$
 1990 LET S=VAL(S$)
 2000 IF RTRIM$(SAENGER$)="DIV" THE
N 2090
 2010 FOR I=1 TO S
 2020   LINE INPUT #106:SONG$(I)
 2030 NEXT I
 2040 GOTO 1910
 2050 CLOSE #106
 2060 RETURN
 2070 GOTO 2020
 2080 CLOSE #106:RETURN
 2090 FOR I=1 TO S
 2100   LINE INPUT #106:SONG$(I)
 2110   LINE INPUT #106:DIV$(I)
 2120 NEXT I
```

```
 2130 GOTO 1910
 2140 !
 2150 ! ****************************
 2160 ! * DATEN-DISKETTE EINLESEN *
 2170 ! ****************************
 2180 !
 2190 CLEAR SCREEN
 2200 !
 2210 PRINT AT 3,10:"HILFSMENUE"
 2220 PRINT AT 4,10:"=========="
 2230 PRINT AT 8,2:"Bitte die nachf
olgenden Fragen mit"
 2240 PRINT AT 9,2:"J(a) oder N(ein
) beantworten !"
 2250 PRINT AT 11,2:"Bitte Daten-Di
skette in Laufwerk 1 !"
 2260 PRINT AT 12,2:"Fertig ? ";
 2270 LET A$=INKEY$
 2280 IF A$="" THEN 2270
 2290 PRINT A$;
 2300 IF A$="J" OR A$="j" THEN 2340
 2310 PING
 2320 GOTO 2260
 2330 !
 2340 PRINT AT 14,2:"Ausgabe auf Dr
ucker umleiten ? ";
 2350 LET A$=INKEY$
 2360 IF A$="" THEN 2350
 2370 PRINT A$;
 2380 IF A$="J" OR A$="j" THEN 2430
 2390 IF A$="N" OR A$="n" THEN 2580
 2400 PING
 2410 GOTO 2340
 2420 !
 2430 PRINT AT 16,2:"Bitte den Druc
ker einschalten. Reicht"
 2440 PRINT AT 17,2:"das Papier noc
h ?"
 2450 PRINT AT 18,2:"Ist der Drucke
r bereit ? ";
 2460 LET A$=INKEY$
 2470 IF A$="" THEN 2460
 2480 PRINT A$
 2490 IF A$="J" OR A$="j" THEN GOTO
 2530
 2500 PING
 2510 GOTO 2450
 2520 !
 2530 PRINT AT 20,2:"Wenn der Druck
vorgang enden soll,"
 2540 PRINT AT 21,2:"bei Anzeige de
s Menüs die 9 wählen."
 2550 !
 2560 LET DR=1 ! DRUCKER-FLAG SETZE
N
 2570 !
 2580 PRINT AT 22,2:"Alle Eingaben
in Großbuchstaben !"
 2590 PRINT AT 23,2:"Weiter ?";
 2600 LET A$=INKEY$
 2610 IF A$="" THEN 2600
 2620 PRINT A$;
 2630 IF A$="J" OR A$="j" THEN 2670
 2640 PING
 2650 GOTO 2580
 2660 !
 2670 ! ******************
 2680 ! * MENUE AUSGABE *
 2690 ! ******************
 2700 !
 2710 CLEAR SCREEN
 2720 !
 2730 PRINT AT 2,9:"SCHALLPLATTEN -
 ARCHIV"
 2740 PRINT AT 3,9:"===============
=======
 2750 PRINT AT 5,2:"Funktionen von
1-8 wählen:"
 2760 PRINT AT 6,1:"---------------
-----------------------"
 2770 PRINT AT 8,2:"Suche LP-Titel
>Eingabe Interpret..1"
 2780 PRINT AT 10,2:"Suche LP-Titel
 >Eingabe Song.......2"
 2790 PRINT AT 12,2:"Suche Song >Ei
ngabe Interpret......3"
 2800 PRINT AT 14,2:"Suche Song >Ei
ngabe LP-Titel.......4"
 2810 PRINT AT 16,2:"Suche Interpre
t >Eingabe Song......5"
 2820 PRINT AT 18,2:"Suche Interpre
t >Eingabe LP-Titel..6"
 2830 PRINT AT 20,2:"Hilfsmenü.....
.....................7"
 2840 PRINT AT 22,2:"Zurück zum Hau
pt-Menü.............8";
 2850 !
 2860 LET A$=INKEY$
 2870 IF A$="" THEN 2860
 2880 LET A=VAL(A$)
 2890 ON A GOTO 2980,3790,4600,5440
,6270,7080,2150,250,2930
 2900 PING
 2910 GOTO 2670
 2920 !
 2930 LET DR=0
 2940 PING
```

```
 2950 PING
 2960 GOTO 2860
 2970 !
 2980 ! ************************
 2990 ! * SUCHEN VON LP-TITEL *
 3000 ! ************************
 3010 !
 3020 CLEAR SCREEN
 3030 !
 3040 !
 3050 PRINT AT 1,8:"Suchen von  LP-
Titel"
 3060 PRINT AT 2,8:"==============
====="
 3070 PRINT AT 3,2:"Eingabe des Int
erpreten"
 3080 LINE INPUT AT 5,2:E$
 3090 IF E$="" THEN 3080
 3100 PRINT AT 6,2:"Folgende LP-Tit
el zu ";E$
 3110 PRINT AT 7,2:"gefunden :"
 3120 PRINT AT 8,2:"--------------
----------------------"
 3130 LET T=0:LET F=0
 3140 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
 3150 !
 3160 LINE INPUT #106,IF MISSING 34
30:NUMMER$
 3170 LET NUMMER=VAL(NUMMER$)
 3180 LINE INPUT #106:EINGABEDATUM$

 3190 LINE INPUT #106:DATUM$
 3200 LINE INPUT #106:SAENGER$
 3210 LINE INPUT #106:TITEL$
 3220 LINE INPUT #106:S$
 3230 LET S=VAL(S$)
 3240 IF RTRIM$(SAENGER$)="DIV" THE
N GOTO 3580
 3250 FOR I=1 TO S
 3260   LINE INPUT #106:SONG$(I)
 3270 NEXT I
 3280 IF RTRIM$(SAENGER$)<>E$ THEN
GOTO 3150
 3290 LET T=T+1
 3300 PRINT AT 8+T,1:NUMMER;TAB(9);
DATUM$;TAB(16);TITEL$
 3310 LET F=1
 3320 IF T<=11 THEN GOTO 3150
 3330 IF DR=1 THEN GOSUB 7910
 3340 PRINT AT 23,2:"WEITER (J/N)?
";
 3350 LET A$=INKEY$
 3360 IF A$="" THEN 3350
 3370 PRINT A$
 3380 IF A$="J" OR A$="j" THEN 3420
 3390 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 2670
 3400 PING
 3410 GOTO 3340
 3420 LET T=0:GOTO 3150
 3430 IF F=0 THEN 3480
 3440 IF DR=1 THEN GOSUB 7910
 3450 !
 3460 PRINT AT 22,2:"Keine weiteren
 Titel gefunden !"
 3470 GOTO 3500
 3480 !
 3490 PRINT AT 22,2:"Keine weiteren
 LP's gefunden !"
 3500 PRINT AT 23,2:"Zurück zum Men
ü (J/N) ? ";
 3510 LET A$=INKEY$
 3520 IF A$="" THEN 3510
 3530 PRINT A$
 3540 IF A$="J" OR A$="j" THEN CLOS
E #106:GOTO 2670
 3550 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 2980
 3560 PING
 3570 GOTO 3510
 3580 FOR I=1 TO S
 3590   LINE INPUT #106:SONG$(I)
 3600   LINE INPUT #106:DIV$(I)
 3610   IF RTRIM$(DIV$(I))<>E$ THEN
 3760
 3620   PRINT AT 9+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);TITEL$
 3630   LET T=T+1:LET F=1
 3640   IF T<=11 THEN 3760
 3650   IF DR=1 THEN GOSUB 7910
 3660   PRINT AT 23,2:"WEITER (J/N)
? ";
 3670   LET A$=INKEY$
 3680   IF A$="" THEN 3670
 3690   PRINT A$
 3700   IF A$="J" OR A$="j" THEN 37
40
 3710   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 3720   PING
 3730   GOTO 3660
 3740   !
 3750   LET T=0
 3760 NEXT I
 3770 GOTO 3150
 3780 !
 3790 ! ************************
 3800 ! * SUCHEN VON LP-TITEL *
 3810 ! ************************
 3820 !
 3830 CLEAR SCREEN
 3840 !
 3850 PRINT AT 1,8:"Suchen von  LP-
```

```
Titel"
 3860 PRINT AT 2,8:"==============
=====
 3870 PRINT AT 3,2:"Bitte den LP-So
ng";
 3880 LINE INPUT AT 5,2:E$
 3890 IF E$="" THEN 3880
 3900 PRINT AT 6,2:"Folgende LP-Tit
el wurden zu ";E$
 3910 PRINT AT 7,2:"gefunden :"
 3920 PRINT AT 8,2:"--------------
----------------------"
 3930 LET T=0:LET F=0:LET Z=0
 3940 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
 3950 !
 3960 LINE INPUT #106,IF MISSING 42
40:NUMMER$
 3970 LET NUMMER=VAL(NUMMER$)
 3980 LINE INPUT #106:EINGABEDATUM$

 3990 LINE INPUT #106:DATUM$
 4000 LINE INPUT #106:SAENGER$
 4010 LINE INPUT #106:TITEL$
 4020 LINE INPUT #106:S$
 4030 LET S=VAL(S$)
 4040 IF RTRIM$(SAENGER$)="DIV" THE
N 4390
 4050 FOR I=1 TO S
 4060   LINE INPUT #106:SONG$(I)
 4070   IF RTRIM$(SONG$(I))=E$ THEN
 LET Z=1
 4080 NEXT I
 4090 IF Z=1 THEN 4110
 4100 GOTO 3950
 4110 PRINT AT 9+T,1:NUMMER;TAB(9);
DATUM$;TAB(16);TITEL$
 4120 LET F=1:LET Z=0:LET T=T+1
 4130 IF T<=11 THEN 3950
 4140 IF DR=1 THEN GOSUB 7910
 4150 PRINT AT 23,2:"WEITER (J/N)?
";
 4160 LET A$=INKEY$
 4170 IF A$="" THEN 4160
 4180 PRINT A$
 4190 IF A$="J" OR A$="j" THEN 4230

 4200 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 2670
 4210 PING
 4220 GOTO 4150
 4230 LET T=0:GOTO 3950
 4240 IF F=0 THEN 4290
 4250 IF DR=1 THEN GOSUB 7910
 4260 !
 4270 PRINT AT 22,2:"Keine weiteren
 Titel gefunden !"
 4280 GOTO 4310
 4290 !
 4300 PRINT AT 22,2:"Zum Song keine
 weiteren LP's !"
 4310 PRINT AT 23,2:"Zurück zum Men
ü (J/N) ? ";
 4320 LET A$=INKEY$
 4330 IF A$="" THEN 4320
 4340 PRINT A$
 4350 IF A$="J" OR A$="j" THEN CLOS
E #106:GOTO 2670
 4360 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 3790
 4370 PING
 4380 GOTO 4320
 4390 FOR I=1 TO S
 4400   LINE INPUT #106:SONG$(I)
 4410   LINE INPUT #106:DIV$(I)
 4420   IF RTRIM$(SONG$(I))<>E$ THE
N 4570
 4430   PRINT AT 9+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);TITEL$
 4440   LET T=T+1:LET F=1
 4450   IF T<=11 THEN 4570
 4460   IF DR=1 THEN GOSUB 7910
 4470   PRINT AT 23,2:"WEITER (J/N)
? ";
 4480   LET A$=INKEY$
 4490   IF A$="" THEN 4480
 4500   PRINT A$
 4510   IF A$="J" OR A$="j" THEN 45
50
 4520   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 4530   PING
 4540   GOTO 4470
 4550   !
 4560   LET T=0
 4570 NEXT I
 4580 GOTO 3950
 4590 !
 4600 ! **********************
 4610 ! * SUCHEN VON SONGS *
 4620 ! **********************
 4630 !
 4640 CLEAR SCREEN
 4650 !
 4660 !
 4670 PRINT AT 1,8:"Suchen von  LP-
Songs"
 4680 PRINT AT 2,8:"==============
=====
 4690 PRINT AT 3,2:"Bitte den Inter
preten"
 4700 LINE INPUT AT 5,2:E$
 4710 IF E$="" THEN 4700
 4720 PRINT AT 6,2:"Folgende LP-Son
```

```
gs zu ";E$
 4730 PRINT AT 7,2:"gefunden :"
 4740 PRINT AT 8,2:"--------------
----------------------"
 4750 LET T=0:LET F=0
 4760 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
 4770 !
 4780 LINE INPUT #106,IF MISSING 50
80:NUMMER$
 4790 LET NUMMER=VAL(NUMMER$)
 4800 LINE INPUT #106:EINGABEDATUM$

 4810 LINE INPUT #106:DATUM$
 4820 LINE INPUT #106:SAENGER$
 4830 LINE INPUT #106:TITEL$
 4840 LINE INPUT #106:S$
 4850 LET S=VAL(S$)
 4860 IF RTRIM$(SAENGER$)="DIV" THE
N 5230
 4870 FOR I=1 TO S
 4880   LINE INPUT #106:SONG$(I)
 4890 NEXT I
 4900 IF RTRIM$(SAENGER$)<>E$ THEN
4770
 4910 FOR P=1 TO S
 4920   LET T=T+1
 4930   PRINT AT 8+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);SONG$(P)
 4940   LET F=1
 4950   IF T<=11 THEN 5060
 4960   IF DR=1 THEN GOSUB 7910
 4970   PRINT AT 23,2:"WEITER (J/N)
? ";
 4980   LET A$=INKEY$
 4990   IF A$="" THEN 4980
 5000   PRINT A$
 5010   IF A$="J" OR A$="j" THEN 50
50
 5020   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 5030   PING
 5040   GOTO 4970
 5050   LET T=0
 5060 NEXT P
 5070 GOTO 4770
 5080 IF F=0 THEN 5130
 5090 IF DR=1 THEN GOSUB 7910
 5100 !
 5110 PRINT AT 22,2:"Keine weiteren
 Songs gefunden !"
 5120 GOTO 5150
 5130 !
 5140 PRINT AT 22,2:"Zum Interprete
n keine weiteren Songs!"
 5150 PRINT AT 23,2:"Zurück zum Men
ü (J/N) ? ";
 5160 LET A$=INKEY$
 5170 IF A$="" THEN 5160
 5180 PRINT A$
 5190 IF A$="J" OR A$="j" THEN CLOS
E #106:GOTO 2670
 5200 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 4600
 5210 PING
 5220 GOTO 5160
 5230 FOR I=1 TO S
 5240   LINE INPUT #106:SONG$(I)
 5250   LINE INPUT #106:DIV$(I)
 5260   IF RTRIM$(DIV$(I))<>E$ THEN
 5410
 5270   PRINT AT 9+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);SONG$(I)
 5280   LET T=T+1:LET F=1
 5290   IF T<=11 THEN 5410
 5300   IF DR=1 THEN GOSUB 7910
 5310   PRINT AT 23,2:"WEITER (J/N)
? ";
 5320   LET A$=INKEY$
 5330   IF A$="" THEN 5320
 5340   PRINT A$
 5350   IF A$="J" OR A$="j" THEN 53
90
 5360   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 5370   PING
 5380   GOTO 5310
 5390   !
 5400   LET T=0
 5410 NEXT I
 5420 GOTO 4770
 5430 !
 5440 ! **********************
 5450 ! * SUCHEN VON SONGS *
 5460 ! **********************
 5470 !
 5480 CLEAR SCREEN
 5490 !
 5500 PRINT AT 1,8:"Suchen von  LP-
Songs"
 5510 PRINT AT 2,8:"==============
======"
 5520 PRINT AT 3,2:"Bitte den LP-Ti
tel"
 5530 LINE INPUT AT 5,2:E$
 5540 IF E$="" THEN 5530
 5550 PRINT AT 6,2:"Folgende LP-Son
gs zu ";E$
 5560 PRINT AT 7,2:"gefunden :"
 5570 PRINT AT 8,2:"--------------
----------------------"
 5580 LET T=0:LET F=0
 5590 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
```

```
 5600 !
 5610 LINE INPUT #106,IF MISSING 59
10:NUMMER$
 5620 LET NUMMER=VAL(NUMMER$)
 5630 LINE INPUT #106:EINGABEDATUM$

 5640 LINE INPUT #106:DATUM$
 5650 LINE INPUT #106:SAENGER$
 5660 LINE INPUT #106:TITEL$
 5670 LINE INPUT #106:S$
 5680 LET S=VAL(S$)
 5690 IF RTRIM$(SAENGER$)="DIV" THE
N 6060
 5700 FOR I=1 TO S
 5710   LINE INPUT #106:SONG$(I)
 5720 NEXT I
 5730 IF RTRIM$(TITEL$)<>E$ THEN 56
00
 5740 FOR P=1 TO S
 5750   LET T=T+1
 5760   PRINT AT 8+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);SONG$(P)
 5770   LET F=1
 5780   IF T<=11 THEN 5890
 5790   IF DR=1 THEN GOSUB 7910
 5800   PRINT AT 23,2:"WEITER (J/N)
? ";
 5810   LET A$=INKEY$
 5820   IF A$="" THEN 5810
 5830   PRINT A$
 5840   IF A$="J" OR A$="j" THEN 58
80
 5850   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 5860   PING
 5870   GOTO 5800
 5880   LET T=1
 5890 NEXT P
 5900 GOTO 5600
 5910 IF F=0 THEN 5960
 5920 IF DR=1 THEN GOSUB 7910
 5930 !
 5940 PRINT AT 22,2:"Keine weiteren
 Songs zum LP-Titel!"
 5950 GOTO 5980
 5960 !
 5970 PRINT AT 22,2:"Zum LP-Titel k
eine Songs gefunden !"
 5980 PRINT AT 23,2:"Zurück zum Men
ü (J/N) ? ";
 5990 LET A$=INKEY$
 6000 IF A$="" THEN 5990
 6010 PRINT A$
 6020 IF A$="J" OR A$="j" THEN CLOS
E #106:GOTO 2670
 6030 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 5440
6040 PING
6050 GOTO 5990
6060 FOR I=1 TO S
6070   LINE INPUT #106:SONG$(I)
6080   LINE INPUT #106:DIV$(I)
6090   IF RTRIM$(TITEL$)<>E$ THEN
6240
 6100   PRINT AT 9+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);SONG$(I)
 6110   LET T=T+1:LET F=1
 6120   IF T<=11 THEN 6240
 6130   IF DR=1 THEN GOSUB 7910
 6140   PRINT AT 23,2:"WEITER (J/N)
? ";
 6150   LET A$=INKEY$
 6160   IF A$="" THEN 6150
 6170   PRINT A$
 6180   IF A$="J" OR A$="j" THEN 62
20
 6190   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 6200   PING
 6210   GOTO 6140
 6220   !
 6230   LET T=0
 6240 NEXT I
 6250 GOTO 5600
 6260 !
 6270 ! ***************************
 6280 ! * SUCHEN VON INTERPRETEN *
 6290 ! ***************************
 6300 !
 6310 CLEAR SCREEN
 6320 !
 6330 PRINT AT 1,8:"Suchen von Inte
rpreten"
 6340 PRINT AT 2,8:"==============
========"
 6350 PRINT AT 3,2:"Bitte den Song
eingeben"
 6360 LINE INPUT AT 5,2:E$
 6370 IF E$="" THEN 6360
 6380 PRINT AT 6,2:"Folgender Inter
pret zu ";E$
 6390 PRINT AT 7,2:"gefunden :"
 6400 PRINT AT 8,2:"--------------
----------------------"
 6410 LET T=0:LET F=0:LET Z=0
 6420 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
 6430 !
 6440 LINE INPUT #106,IF MISSING 67
20:NUMMER$
 6450 LET NUMMER=VAL(NUMMER$)
 6460 LINE INPUT #106:EINGABEDATUM$

 6470 LINE INPUT #106:DATUM$
```

```
 6480 LINE INPUT #106:SAENGER$
 6490 LINE INPUT #106:TITEL$
 6500 LINE INPUT #106:S$
 6510 LET S=VAL(S$)
 6520 IF RTRIM$(SAENGER$)="DIV" THE
N 6870
 6530 FOR I=1 TO S
 6540   LINE INPUT #106:SONG$(I)
 6550   IF RTRIM$(SONG$(I))=E$ THEN
 LET Z=1
 6560 NEXT I
 6570 IF Z=1 THEN 6590
 6580 GOTO 6430
 6590 PRINT AT 9+T,1:NUMMER;TAB(9);
DATUM$;TAB(16);SAENGER$
 6600 LET F=1:LET Z=0:LET T=T+1
 6610 IF T<=11 THEN 6430
 6620 IF DR=1 THEN GOSUB 7910
 6630 PRINT AT 23,2:"WEITER (J/N)?
";
 6640 LET A$=INKEY$
 6650 IF A$="" THEN 6640
 6660 PRINT A$
 6670 IF A$="J" OR A$="j" THEN 6710

 6680 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 2670
 6690 PING
 6700 GOTO 6630
 6710 LET T=0:GOTO 6430
 6720 IF F=0 THEN 6770
 6730 IF DR=1 THEN GOSUB 7910
 6740 !
 6750 PRINT AT 22,2:"Kein weiterer
Interpret zum Song!"
 6760 GOTO 6790
 6770 !
 6780 PRINT AT 22,2:"Zum Song kein
weiterer Interpret !"
 6790 PRINT AT 23,2:"Zurück zum Men
ü (J/N) ? ";
 6800 LET A$=INKEY$
 6810 IF A$="" THEN 6800
 6820 PRINT A$
 6830 IF A$="J" OR A$="j" THEN CLOS
E #106:GOTO 2670
 6840 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 6270
 6850 PING
 6860 GOTO 6800
 6870 FOR I=1 TO S
 6880   LINE INPUT #106:SONG$(I)
 6890   LINE INPUT #106:DIV$(I)
 6900   IF RTRIM$(SONG$(I))<>E$ THE
N 7050
 6910   PRINT AT 9+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);DIV$(I)
 6920   LET T=T+1:LET F=1
 6930   IF T<=11 THEN 7050
 6940   IF DR=1 THEN GOSUB 7910
 6950   PRINT AT 23,2:"WEITER (J/N)
? ";
 6960   LET A$=INKEY$
 6970   IF A$="" THEN 6960
 6980   PRINT A$
 6990   IF A$="J" OR A$="j" THEN 70
30
 7000   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 7010   PING
 7020   GOTO 6950
 7030   !
 7040   LET T=0
 7050 NEXT I
 7060 GOTO 6430
 7070 !
 7080 ! ******************************
 7090 ! * SUCHEN VON INTERPRETEN *
 7100 ! ******************************
 7110 !
 7120 CLEAR SCREEN
 7130 !
 7140 PRINT AT 1,8:"Suchen von Inte
rpreten"
 7150 PRINT AT 2,8:"==============
========"
 7160 PRINT AT 3,2:"Bitte den LP-Ti
tel eingeben"
 7170 LINE INPUT AT 5,2:E$
 7180 IF E$="" THEN 7170
 7190 PRINT AT 6,2:"Folgender Inter
pret zu ";E$
 7200 PRINT AT 7,2:"gefunden :"
 7210 PRINT AT 8,2:"--------------
----------------------"
 7220 LET T=0:LET F=0
 7230 OPEN #106:"disk:LPDATEI.DAT"
ACCESS INPUT
 7240 !
 7250 LINE INPUT #106,IF MISSING 75
20:NUMMER$
 7260 LET NUMMER=VAL(NUMMER$)
 7270 LINE INPUT #106:EINGABEDATUM$

 7280 LINE INPUT #106:DATUM$
 7290 LINE INPUT #106:SAENGER$
 7300 LINE INPUT #106:TITEL$
 7310 LINE INPUT #106:S$
 7320 LET S=VAL(S$)
 7330 IF RTRIM$(SAENGER$)="DIV" THE
N 7670
 7340 FOR I=1 TO S
 7350   LINE INPUT #106:SONG$(I)
 7360 NEXT I
```

```
 7370 IF RTRIM$(TITEL$)<>E$ THEN 72
40
 7380 LET T=T+1
 7390 PRINT AT 8+T,1:NUMMER;TAB(9);
DATUM$;TAB(16);SAENGER$
 7400 LET F=1
 7410 IF T<=11 THEN 7240
 7420 IF DR=1 THEN GOSUB 7910
 7430 PRINT AT 23,2:"WEITER (J/N)?
";
 7440 LET A$=INKEY$
 7450 IF A$="" THEN 7440
 7460 PRINT A$
 7470 IF A$="J" OR A$="j" THEN 7510

 7480 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 2670
 7490 PING
 7500 GOTO 7430
 7510 LET T=0:GOTO 7240
 7520 IF F=0 THEN 7570
 7530 IF DR=1 THEN GOSUB 7910
 7540 !
 7550 PRINT AT 22,2:"Kein weiterer
Interpret zum LP-Titel!"
 7560 GOTO 7590
 7570 !
 7580 PRINT AT 22,2:"Zum LP-Titel k
ein weiterer Interpret!"
 7590 PRINT AT 23,2:"Zurück zum Men
ü (J/N) ? ";
 7600 LET A$=INKEY$
 7610 IF A$="" THEN 7600
 7620 PRINT A$;
 7630 IF A$="J" OR A$="j" THEN CLOS
E #106:GOTO 2670
 7640 IF A$="N" OR A$="n" THEN CLOS
E #106:GOTO 7080
 7650 PING
 7660 GOTO 7600
 7670 FOR I=1 TO S
 7680   LINE INPUT #106:SONG$(I)
 7690   LINE INPUT #106:DIV$(I)
 7700 NEXT I
 7710 IF RTRIM$(TITEL$)<>E$ THEN 72
40
 7720 FOR P=1 TO S
 7730   LET T=T+1
 7740   PRINT AT 8+T,1:NUMMER;TAB(9
);DATUM$;TAB(16);DIV$(P)
 7750   LET F=1
 7760   IF T<=11 THEN 7880
 7770   IF DR=1 THEN GOSUB 7910
 7780   PRINT AT 23,2:"WEITER (J/N)
? ";
 7790   LET A$=INKEY$
 7800   IF A$="" THEN 7790
 7810   PRINT A$
 7820   IF A$="J" OR A$="j" THEN 78
60
 7830   IF A$="N" OR A$="n" THEN CL
OSE #106:GOTO 2670
 7840   PING
 7850   GOTO 7780
 7860   !
 7870   LET T=0
 7880 NEXT P
 7890 GOTO 7240
 7900 !
 7910 ! ************
 7920 ! * HARDCOPY *
 7930 ! ************
 7940 !
 7950 COPY
 7960 RETURN
 7970 !
 7980 ! ***************************
 7990 ! * UMBENENNEN DER DATEI *
 8000 ! ***************************
 8010 !
 8020 EXT "COPY A:LPDATEI.DAT A:LPD
ATEI.BAK"
 8030 GOTO 8040
 8040 !
 8050 EXT "ERA A:LPDATEI.DAT"
 8060 GOTO 8070
 8070 !
 8080 RETURN
 8090 !
 8100 ! ******************************

 8110 ! * APPEND AN 'LPDATEI.DAT' *

 8120 ! ******************************

 8130 !
 8140 OPEN #121:"disk:LPDATEI.BAK"
ACCESS INPUT
 8150 !
 8160 LINE INPUT #121,IF MISSING 81
90:DUMMY$
 8170 PRINT #106:DUMMY$
 8180 GOTO 8150
 8190 CLOSE #121
 8200 EXT "ERA A:LPDATEI.BAK"
 8210 GOTO 8220
 8220 !
 8230 RETURN
 8240 !
 8250 ! ************************
 8260 ! * Mache SPACE-String *
 8270 ! ************************
 8280 !
 8290 DEF SPC(LENGTH)
```

```
8300    NUMERIC ZAEHLER
8310    !
8320    FOR ZAEHLER=1 TO LENGTH
8330      PRINT CHR$(32);
8340    NEXT ZAEHLER
8350    !
8360  END DEF
```

Polynome

Rechner: Enterprise 64/128
Programmname: Polynome
Programmlänge: ca. 12 300 Bytes
Programmiersprache: BASIC
Zubehör: Drucker

Funktionserstellung

Der Volksmund sagt: "Glaub' keiner Statistik, die Du nicht selbst gefälscht hast". Eine Möglichkeit hierzu bietet Ihnen folgendes Programm, mit dem Sie anhand von gegebenen Punkten eine Funktion erstellen lassen können.
Zur Bedienung: Nach Starten des Programms erscheint das Hauptmenü.

Eingabe von Werten

* Grad des Polynoms
  Falls Sie ein Polynom als Funktion errechnet haben wollen, so müssen Sie hier dessen Grad angeben; für die übrigen Funktionen ist diese Eingabe ohne Bedeutung (also unnötig).
* Eingabe der Kurvenpunkte
  Hier geben Sie Ihre zunächst die Anzahl Ihrer Werte an. Nun frägt das Programm die einzelnen Wertepaare ab. Aber Achtung: Der Grad des Polynoms darf maximal gleich der Anzahl der Punkte sein!
* Rückkehr zum Hauptmenü
  Damit schließen Sie vorläufig Ihre Eingabe ab (erneuter Aufruf ist jederzeit möglich).

Polynom

Sobald Sie dieses Unterprogramm aufrufen, wird ein Polynom n-ten Grades erstellt. Dabei ist n beliebig (maximal 10, ansonsten ist eine Änderung in Zeile 210 "A1=..."). Dabei wird das Prinzip der Gauss-'schen Elimination angewendet. Dies bedeutet, daß die eingegebenen Punkte als Matrix abgelegt werden. Auch für die Koeffizienten des Polynoms wird eine Matrix gebildet (n*n). Durch geschickte Umformungen wird das Gleichungssystem dann gelöst.
Ist die Berechnung fertig, so ertönt ein Klingelzeichen. Hierauf haben Sie zur Ausgabe der Daten die Wahl zwischen Drucker und Bildschirm. Nach Abschluß der Ausgabe kehrt das Programm zum Hauptmenü zurück.

Potenzfunktion

Diese Funktion verläuft analog zum "Polynom", allerdings mit dem Unterschied, daß der Exponent beliebig ist (bei Polynom 0<=i<=n, i ist dabei eine natürliche Zahl).
Auch hierauf verzweigt das Programm zuerst zur Ausgabe und daraufhin ins Hauptmenü.

Exponentialfunktion

Hier wird eine Exponentialfunktion der Form f(x)=c(1)*EXP(c(2)*x) erstellt. Diese ist für Wachstumsprozesse gut geeignet, bei linearen Vorgängen liefert sie eine eher grobe Näherung. Ist die Berechnung fertig, verfährt der Rechner wie unter "Polynome".

Funktion plotten

Falls Sie eine Funktion berechnet haben (egal was für einen Typ), so können Sie diese zeichnen lassen. Dafür benötigt der Computer die Grenzen für die x- und y-Werte. Anhand dieser Angaben berechnet das

Programm die Maßstäbe für die Darstellung sowie die Lage des Ursprungs. Nun wird ein Koordinaten-Gitter aufgebaut, in dem sowohl die Funktion als auch die vorgegebenen Punkte eingetragen werden. Erfolgt nun ein Tastendruck, so befinden Sie sich wieder im Menü.

Werte-Ausgabe

Unter diesem Menüpunkt ist es möglich, die letzten Daten in Form einer Tabelle auszugeben (wahlweise auf den Bildschirm oder Drucker).

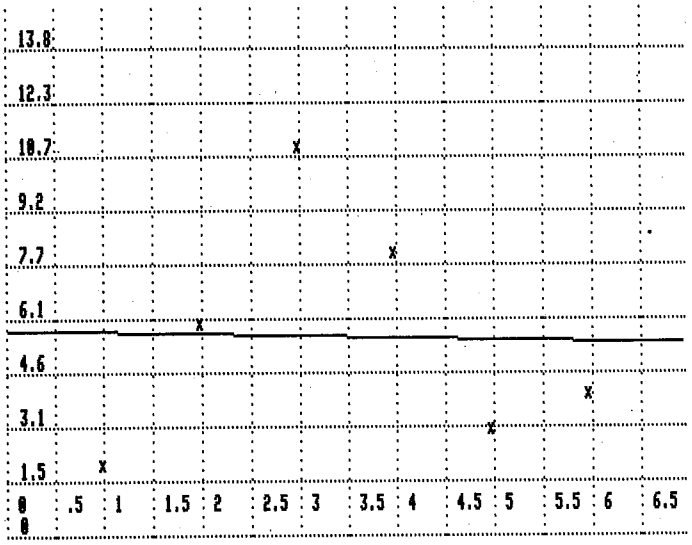
Programmende

Hiermit werden alle Fenster geschlossen und der 40-Zeichen -Modus wiederhergestellt.

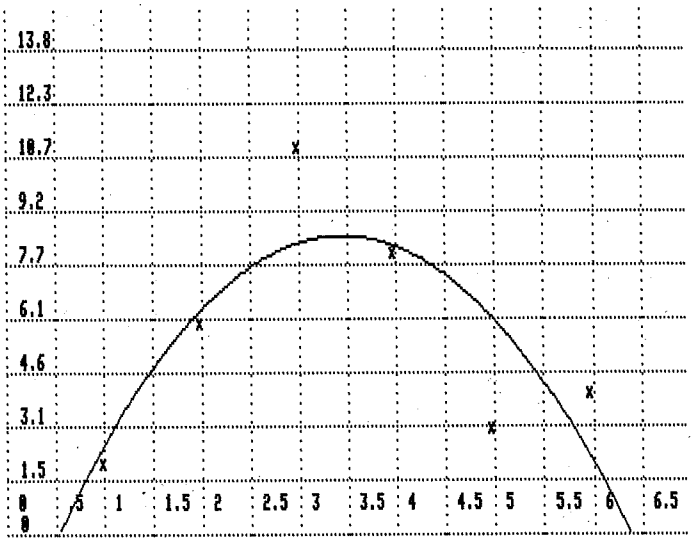
Thomas Müller

Die nebenstehenden Hardcopies zeigen die Annäherung einer Kurve an gegebene Werte.
Für die nachfolgenden Diagramme wurden folgende Eingaben getätigt:
P1 = (1/2)
P2 = (2/6)
P3 = (3/11)
P4 = (4/8)
P5 = (5/3)
P6 = (6/4)
Bei den Abbildungen stehen jeweils der Typ der Funktion sowie die Summe der Fehlerquadrate.

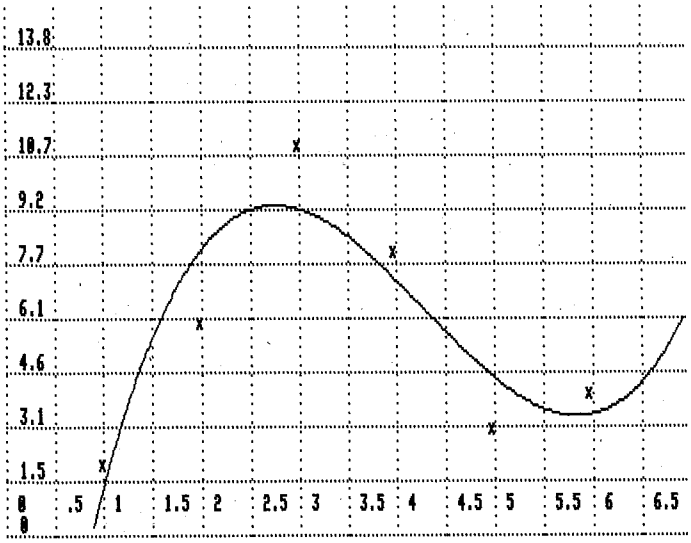
Polynom 1.Grades: 57.27619047



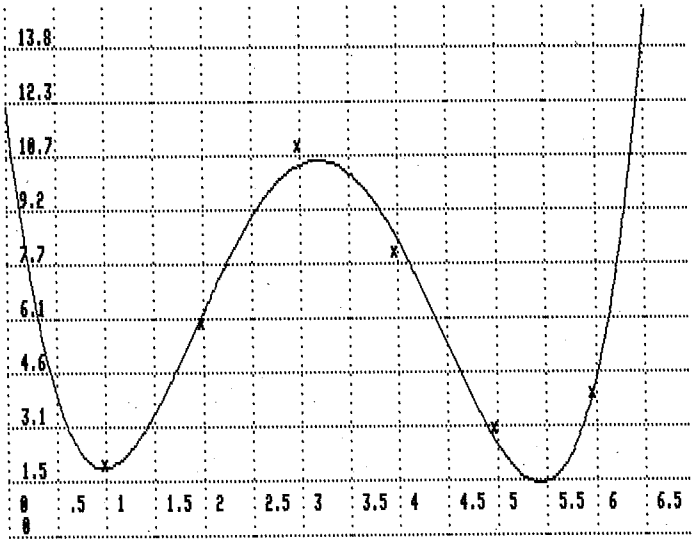
Polynom 2.Grades: 21.26428568



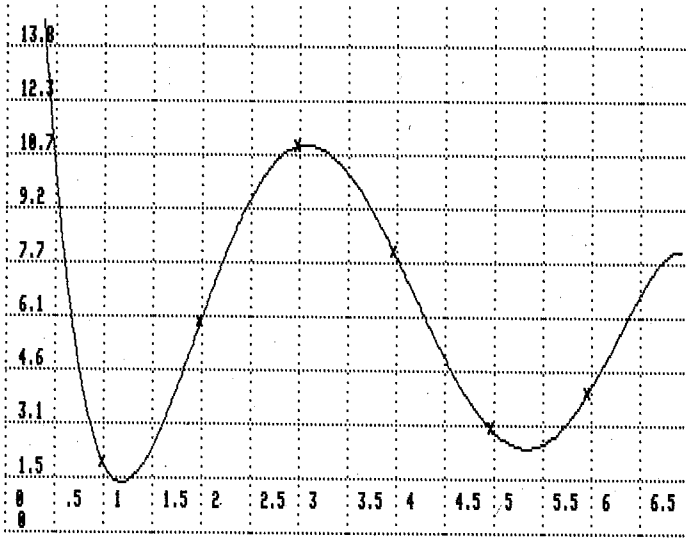
Polynom 3.Grades: 10.9920635



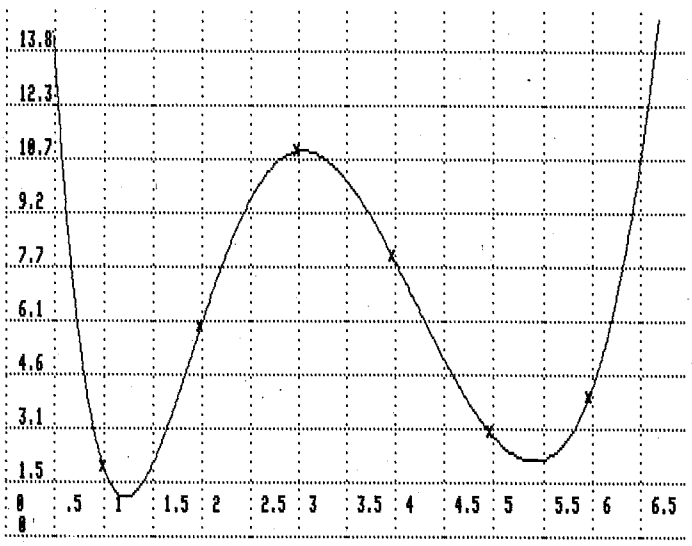
Polynom 4.Grades: .6706349567



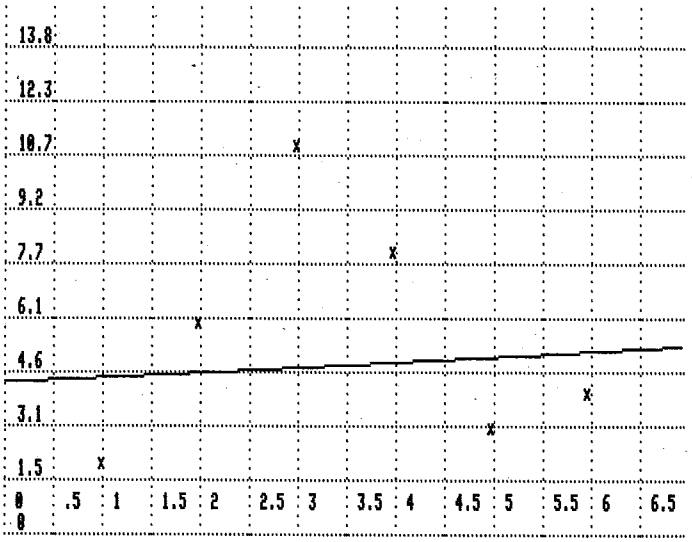
Polynom 5.Grades: 1.165887762 E-06



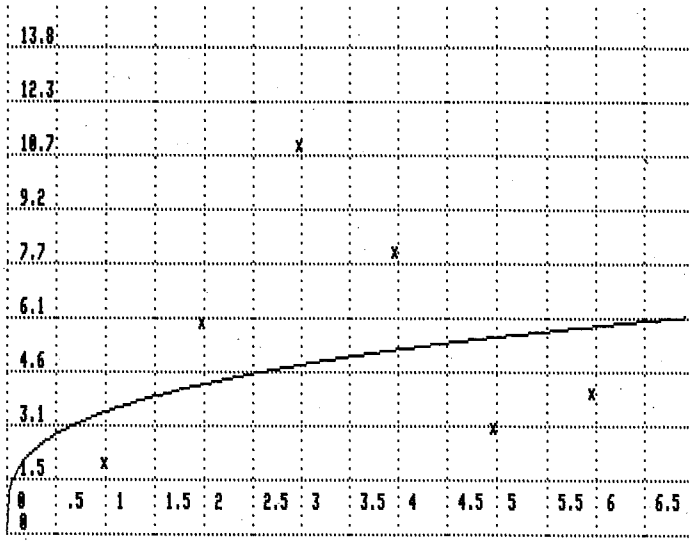
Polynom 6.Grades: 1.088188437 E-07



Potenzfunktion: 61.328264



Exponentialfkt.: 62.32242153




```
  100 !Polynomerstellung
  110 !für Chip Special
  120 !von Thomas Müller
  130 !
  140 GOTO 180
  150 CALL MENUE
  160 END
  170 !
  180 !**** Init ****
  190 !
  200 CLEAR SCREEN:SET STATUS OFF:T
EXT 40
  210 LET A1=10:LET A2=20
  220 DIM A(A1,A1),B(A1,A1),C(A1),D
(A1),L(A1)
  230 DIM X1(A2),X2(A2),Y1(A2),Y2(A
2),Y3(A2),UP$(6),US$(2)
  240 !--- Unterprogramm-Namen ---
  250 FOR I=0 TO 6
  260   READ UP$(I)
  270 NEXT I
  280 DATA "Werte eingeben","Polyno
m","Potenzfunktion","Exponentialfun
ktion"
  290 DATA "Funktion plotten","Wert
e-Ausgabe","Programm beenden"
  300 LET AS,FF,D1,N1,N2,N3,H1,H2,H
3,F,FS,DF,XI,XS,YI,YS,UX,UY,MX,MY=0

  310 LET US$(1)="-####.##########"

  320 LET US$(2)="+####.##########"

  330 FOR I=1 TO A2
```

```
  340    LET X2(I),Y2(I)=0
  350  NEXT I
  360  CALL CLEARV
  370  SET #102:PALETTE BLACK,BLUE,R
ED,0,0,0,0,0
  380  GOTO 150
  390  !
  400  !**** UP clear variables ****

  410  !
  420  DEF CLEARV
  430    FOR I=0 TO A1
  440      FOR I2=0 TO A1
  450        LET A(I,I2),B(I,I2)=0
  460      NEXT I2
  470    NEXT I
  480    FOR I=1 TO A2
  490      LET X1(I)=X2(I):LET Y1(I)
=Y2(I)
  500    NEXT I
  510    FOR I=0 TO A1
  520      LET D(I)=0
  530    NEXT I
  540  END DEF
  550  !
  560  !**** Menü ****
  570  !
  580  DEF MENUE
  590    CALL HEAD
  600    !--- Graphikseite 2 ---
  610    SET VIDEO MODE 5
  620    SET VIDEO COLOUR 0
  630    SET VIDEO X 40
  640    SET VIDEO Y 19
  650    OPEN #2:"video:"
  660    DISPLAY #2:AT 6 FROM 1 TO 1
9
  670    CLEAR #2
  680    SET #2:CURSOR CHARACTER 32
  690    SET #2:CHARACTER 140,192,22
4,240,248,252,248,240,224,192
  700    !--- Menü ---
  710    PLOT #2:576,630,
  720    PRINT #2:TAB(18);"Menü:"
  730    PRINT #2:TAB(18);"====="
  740    PLOT #2:320,500,
  750    PRINT #2:
  760    FOR I=0 TO 6
  770      PRINT #2:"            ";UP
$(I)
  780      PRINT #2:
  790    NEXT I
  800    !--- Wahl ---
  810    LET I,G=0
  820    SET #2:INK 0
  830    PLOT #2:300,468-G*71,
  840    PRINT #2:CHR$(159);
  850    SET #2:INK 1
  860    PLOT #2:300,468-I*71,
  870    PRINT #2:CHR$(140);
  880    LET A$=INKEY$
  890    IF A$="" THEN 880
  900    LET AS=ORD(A$)
  910    IF AS=176 THEN LET G=I:LET
I=I-SGN(I):GOTO 820
  920    IF AS=180 THEN LET G=I:LET
I=I+SGN(6-I):GOTO 820
  930    IF AS<>32 THEN 880
  940    LET AS=I:CLEAR #2
  950    PING
  960    !--- Aufruf ---
  970    SELECT CASE AS
  980      !--- Eingabe ---
  990    CASE 0
 1000      CALL EINGABE
 1010      GOTO 670
 1020      !--- Polynom ---
 1030    CASE 1
 1040      CALL CLEARV
 1050      LET N3=N1
 1060      LET F=1
 1070      CALL ELEMENTE
 1080      CALL GAUSS
 1090      IF FF=1 THEN 670
 1100      CALL POL.Y
 1110      CALL AUSGABE
 1120      GOTO 670
 1130      !--- Potenzfunktion ---
 1140    CASE 2
 1150      CALL CLEARV
 1160      LET N3=2
 1170      LET F=2
 1180      CALL LOGX
 1190      CALL LOGY
 1200      CALL ELEMENTE
 1210      CALL GAUSS
 1220      IF FF=1 THEN 670
 1230      CALL POT.Y
 1240      CALL AUSGABE
 1250      GOTO 670
 1260      !--- Exponentialfunktion
---
 1270    CASE 3
 1280      CALL CLEARV
 1290      LET N3=2
 1300      LET F=3
 1310      CALL LOGY
 1320      CALL ELEMENTE
 1330      CALL GAUSS
 1340      CALL EXP.Y
 1350      CALL AUSGABE
 1360      GOTO 670
 1370      !--- Funktion plotten ---
 1380    CASE 4
```

```
 1390      CALL PAR.PLOT
 1400      CALL PLOT
 1410      GOTO 670
 1420      !--- Werte - Ausgabe ---
 1430    CASE 5
 1440      IF FF=1 THEN 670
 1450      CALL AUSGABE
 1460      GOTO 670
 1470      !--- Ende ---
 1480    CASE 6
 1490      CALL ENDE
 1500    END SELECT
 1510 END DEF
 1520 !
 1530 !**** Polynomwerte y() ****
 1540 !
 1550 DEF POL.Y
 1560    FOR I=1 TO N2
 1570      LET Y3(I)=0
 1580      FOR I2=1 TO N1
 1590        LET Y3(I)=Y3(I)+C(I2)*X
2(I)^(I2-1)
 1600      NEXT I2
 1610    NEXT I
 1620 END DEF
 1630 !
 1640 !**** Funktionsplot-Auswahl *
***
 1650 !
 1660 DEF PLOT
 1670    LET TE=2
 1680    SELECT CASE F
 1690      !--- Polynom ---
 1700    CASE 1
 1710      CALL F1_PLOT
 1720      !--- Potenzfunktion ---
 1730    CASE 2
 1740      CALL F2_PLOT
 1750      !--- Exponentialfkt. ---
 1760    CASE 3
 1770      CALL F3_PLOT
 1780    END SELECT
 1790    CALL PUNKTE
 1800    !--- Display-Reset ---
 1810    LET A$=INKEY$
 1820    IF A$="" THEN 1810
 1830    CLOSE #101
 1840    TEXT 40:SET #102:PALETTE BL
ACK,BLUE,RED,0,0,0,0,0
 1850    DISPLAY #2:AT 6 FROM 1 TO 1
9
 1860    CALL HEAD
 1870 END DEF
 1880 !
 1890 !**** Gaussche Elimination **
**
 1900 !
 1910 DEF GAUSS
 1920    LET FF=0
 1930    FOR I=1 TO N3
 1940      LET L(I)=D(I)
 1950      FOR I2=1 TO N3
 1960        LET B(I,I2)=A(I,I2)
 1970      NEXT I2
 1980    NEXT I
 1990    LET FF=0
 2000    FOR I=1 TO N3-1
 2010      LET H1=I
 2020      LET H2=I+1
 2030      LET BB=ABS(B(I,I))
 2040      FOR I2=H2 TO N3
 2050        IF ABS(B(I2,I))>=BB THE
N
 2060          LET BB=ABS(B(I2,I))
 2070          LET H1=I2
 2080        END IF
 2090      NEXT I2
 2100      IF BB=0 THEN
 2110        LET FF=1
 2120        CALL FEHLER
 2130        GOTO 2470
 2140      END IF
 2150      IF H1<>I THEN
 2160        FOR I2=1 TO N3
 2170          LET H3=B(H1,I2)
 2180          LET B(H1,I2)=B(I,I2)
 2190          LET B(I,I2)=H3
 2200        NEXT I2
 2210        LET H3=L(H1)
 2220        LET L(H1)=L(I)
 2230        LET L(I)=H3
 2240      END IF
 2250      FOR I2=H2 TO N3
 2260        LET T=B(I2,I)/B(I,I)
 2270        FOR I3=H2 TO N3
 2280          LET B(I2,I3)=B(I2,I3)
-T*B(I,I3)
 2290        NEXT I3
 2300        LET L(I2)=L(I2)-T*L(I)
 2310      NEXT I2
 2320    NEXT I
 2330    IF B(N3,N3)=0 THEN
 2340      LET FF=1
 2350      CALL FEHLER
 2360      GOTO 2470
 2370    END IF
 2380    LET C(N3)=L(N3)/B(N3,N3)
 2390    !--- Rückwärtsschluß ---
 2400    FOR I=N3-1 TO 1 STEP-1
 2410      LET S=0
 2420      FOR I2=I+1 TO N3
 2430        LET S=S+B(I,I2)*C(I2)
 2440      NEXT I2
 2450      LET C(I)=(L(I)-S)/B(I,I)
```

```
 2460    NEXT I
 2470  END DEF
 2480  !
 2490  !**** Ausgabe ****
 2500  !
 2510  DEF AUSGABE
 2520    CLEAR TEXT:LET DF=0
 2530    DISPLAY #102:AT 6 FROM 6 TO
 24
 2540    INPUT AT 12,8,PROMPT "Ausdr
uck erwünscht (j/n)? ":A$
 2550    IF A$="j" THEN LET DF=1
 2560    SELECT CASE F
 2570      !--- Polynom ---
 2580    CASE 1
 2590      CALL POL.OUT
 2600      !--- Potenzfkt. ---
 2610    CASE 2
 2620      CALL POT.OUT
 2630      !--- Exponentialfkt. ---
 2640    CASE 3
 2650      CALL EXP.OUT
 2660    END SELECT
 2670  END DEF
 2680  !
 2690  !**** Eingabe von Werten ****

 2700  !
 2710  DEF EINGABE
 2720    PLOT #2:300,630,
 2730    PRINT #2:"Eingabe von Werte
n"
 2740    PLOT #2:300,590;876,590
 2750    PLOT #2:1,400,
 2760    PRINT #2:
 2770    PRINT #2:"      1 <= Grad de
s Polynoms"
 2780    PRINT #2:
 2790    PRINT #2:"      2 <= Eingabe
 der Kurvenpunkte"
 2800    PRINT #2:
 2810    PRINT #2:"      3 <= Hauptme
nü"
 2820    LET A$=INKEY$
 2830    IF A$="" THEN 2820
 2840    LET J=VAL(A$)
 2850    IF J<1 OR J>3 THEN 2820
 2860    PING
 2870    SELECT CASE J
 2880      !--- Grad des Polynoms --
-
 2890    CASE 1
 2900      CLEAR TEXT
 2910      DISPLAY #102:AT 6 FROM 6
TO 24
 2920      INPUT AT 12,12,PROMPT "Gr
ad des Polynoms: ":N1
 2930      LET N1=N1+1
 2940      DISPLAY #2:AT 6 FROM 1 TO
 19
 2950      GOTO 2820
 2960      !--- Kurvenpunkte ---
 2970    CASE 2
 2980      CLEAR TEXT
 2990      DISPLAY #102:AT 6 FROM 6
TO 24
 3000      INPUT AT 10,3,PROMPT "Anz
ahl der Wertepaare: ":N2
 3010      PRINT :PRINT
 3020      FOR I=1 TO N2
 3030        PRINT "---------------"
:PRINT I;".tes Paar:"
 3040        INPUT PROMPT "x: ":X1(I
)
 3050        INPUT PROMPT "y: ":Y1(I
)
 3060        LET X2(I)=X1(I)
 3070        LET Y2(I)=Y1(I)
 3080      NEXT I
 3090      DISPLAY #2:AT 6 FROM 1 TO
 19
 3100      GOTO 2820
 3110    CASE 3
 3120    END SELECT
 3130  END DEF
 3140  !
 3150  !**** Programm beenden ****
 3160  !
 3170  DEF ENDE
 3180    CLEAR #2
 3190    CLEAR #101
 3200    CLOSE #2
 3210    CLOSE #101
 3220    TEXT
 3230  END DEF
 3240  !
 3250  !**** logy ****
 3260  !
 3270  DEF LOGY
 3280    FOR I3=1 TO N2
 3290      LET Y1(I3)=LOG(Y1(I3))
 3300    NEXT I3
 3310  END DEF
 3320  !
 3330  !**** logx ****
 3340  !
 3350  DEF LOGX
 3360    FOR I3=1 TO N2
 3370      LET X1(I3)=LOG(X1(I3))
 3380    NEXT I3
 3390  END DEF
 3400  !
 3410  !**** Potenzfkt. y ****
 3420  !
```

```
 3430 DEF POT.Y
 3440   LET C(1)=EXP(C(1))
 3450   FOR I=1 TO N2
 3460     LET Y3(I)=C(1)*X2(I)^C(2)
 3470   NEXT I
 3480 END DEF
 3490 !
 3500 !**** Elemente von A und D ****
 3510 !
 3520 DEF ELEMENTE
 3530   FOR I=1 TO N3
 3540     FOR I2=1 TO N3
 3550       IF I+I2>2 THEN
 3560         FOR I3=1 TO N2
 3570           LET A(I,I2)=A(I,I2)+X1(I3)^(I+I2-2)
 3580         NEXT I3
 3590       ELSE
 3600         LET A(I,I2)=N2
 3610       END IF
 3620     NEXT I2
 3630     FOR I2=1 TO N2
 3640       IF I=1 THEN
 3650         LET D(I)=D(I)+Y1(I2)
 3660       ELSE
 3670         LET D(I)=D(I)+Y1(I2)*X1(I2)^(I-1)
 3680       END IF
 3690     NEXT I2
 3700   NEXT I
 3710 END DEF
 3720 !
 3730 !**** Fehler ****
 3740 !
 3750 DEF FEHLER
 3760   CLEAR TEXT
 3770   DISPLAY #102:AT 6 FROM 6 TO 24
 3780   PRINT AT 10,13:"Fehler:"
 3790   PRINT AT 12,6:"Die Matrix ist singulär!"
 3800   LET A$=INKEY$
 3810   IF A$="" THEN 3800
 3820   DISPLAY #2:AT 6 FROM 1 TO 19
 3830   LET FF=1
 3840 END DEF
 3850 !
 3860 !**** Polynom-Ausgabe ****
 3870 !
 3880 DEF POL.OUT
 3890   CLEAR TEXT
 3900   DISPLAY #102:AT 6 FROM 6 TO 24
 3910   PRINT AT 7,10:"Daten des Polynoms"
 3920   PRINT AT 8,10:"----------"
 3930   CALL STELLEN
 3940   PRINT AT 14,0:"f(x)=";
 3950   PRINT USING US$(1):C(1)
 3960   PRINT AT 15,6,USING US$(2):C(2);
 3970   PRINT "*x"
 3980   FOR I=3 TO N1
 3990     PRINT AT 13+I,6,USING US$(2):C(I);
 4000     PRINT "*x^";(I-1)
 4010   NEXT I
 4020   PRINT
 4030   LET A$=INKEY$
 4040   IF A$="" THEN 4030
 4050   CALL TABELLE
 4060   LET A$=INKEY$
 4070   IF A$="" THEN 4060
 4080   DISPLAY #2:AT 6 FROM 1 TO 19
 4090 END DEF
 4100 !
 4110 !**** Potenzfkt.-Ausgabe ****
 4120 !
 4130 DEF POT.OUT
 4140   CLEAR TEXT
 4150   DISPLAY #102:AT 6 FROM 6 TO 24
 4160   PRINT AT 7,10:"Daten der Potenzfunktion"
 4170   PRINT AT 8,10:"----------"
 4180   CALL STELLEN
 4190   PRINT AT 14,0:"y=";
 4200   PRINT USING US$(1):C(1);
 4210   PRINT "*x^";
 4220   PRINT USING US$(1):C(2)
 4230   LET A$=INKEY$
 4240   IF A$="" THEN 4230
 4250   CALL TABELLE
 4260   LET A$=INKEY$
 4270   IF A$="" THEN 4260
 4280   DISPLAY #2:AT 6 FROM 1 TO 19
 4290 END DEF
 4300 !
 4310 !**** Ausgabe Exponentialfkt. ****
 4320 !
 4330 DEF EXP.OUT
 4340   CLEAR TEXT
 4350   DISPLAY #102:AT 6 FROM 6 TO 24
 4360   PRINT AT 7,6:"Daten der Exponentialfunktion"
```

```
 4370   PRINT AT 8,6:"-------------
----------------"
 4380   CALL STELLEN
 4390   PRINT AT 14,0:"y =       ";
 4400   PRINT USING US$(1):C(1)
 4410   PRINT "    *EXP(";
 4420   PRINT USING US$(1):C(2);
 4430   PRINT "*x)"
 4440   LET A$=INKEY$
 4450   IF A$="" THEN 4440
 4460   CALL TABELLE
 4470   LET A$=INKEY$
 4480   IF A$="" THEN 4470
 4490   DISPLAY #2:AT 6 FROM 1 TO 1
9
 4500 END DEF
 4510 !
 4520 !**** Exponentialfkt. y() ***
*
 4530 !
 4540 DEF EXP.Y
 4550   LET C(1)=EXP(C(1))
 4560   FOR I=1 TO N2
 4570     LET Y3(I)=C(1)*EXP(C(2)*X
2(I))
 4580   NEXT I
 4590 END DEF
 4600 !
 4610 !**** Ausgabeformat ****
 4620 !
 4630 DEF STELLEN
 4640   LET US$(1)="-####.#########
#"
 4650   LET US$(2)="+####.#########
#"
 4660   PRINT AT 10,0:"Auf wieviele
 Nachkommastellen sollen"
 4670   INPUT AT 12,0,PROMPT "die W
erte ausgegeben werden? ":H1
 4680   IF H1>10 THEN 4670
 4690   LET US$(1)=US$(1)(1:6+H1)
 4700   LET US$(2)=US$(2)(1:6+H1)
 4710 END DEF
 4720 !
 4730 !**** Tabelle ****
 4740 !
 4750 DEF TABELLE
 4760   IF DF=1 THEN COPY
 4770   CLEAR TEXT
 4780   PRINT AT 7,16:"Tabelle"
 4790   PRINT AT 8,16:"-------"
 4800   PRINT AT 10,4:"Eingabewerte
   :  Funktionswerte"
 4810   PRINT AT 11,2:"-----------
-----:----------------"
 4820   PRINT TAB(19) ":"
 4830   FOR I=1 TO N2
 4840     PRINT USING US$(1):Y2(I);

 4850     PRINT TAB(19) ":";
 4860     PRINT USING US$(1):Y3(I)
 4870   NEXT I
 4880   CALL FQUADR
 4890   PRINT USING "":
 4900   PRINT :PRINT "     Summe de
r Fehlerquadrate:"
 4910   PRINT :PRINT TAB(12);FS
 4920   IF DF=1 THEN COPY
 4930 END DEF
 4940 !
 4950 !**** Summe der Fehlerquadrat
e ****
 4960 !
 4970 DEF FQUADR
 4980   LET FS=0
 4990   FOR I=1 TO N2
 5000     LET FS=FS+(Y2(I)-Y3(I))^2

 5010   NEXT I
 5020 END DEF
 5030 !
 5040 !**** Funktionsplot-Parameter
 ****
 5050 !
 5060 DEF PAR.PLOT
 5070   CLEAR TEXT
 5080   DISPLAY #102:AT 6 FROM 6 TO
 24
 5090   PRINT AT 7,14:"Funktionsplo
t"
 5100   PRINT AT 8,14:"-----------
-"
 5110   PRINT AT 10,5:"Parameter fü
r die Darstellung:"
 5120   INPUT AT 13,5,PROMPT "Unter
e x-Grenze: ":XI
 5130   INPUT AT 15,5,PROMPT "Obere
 x-Grenze:  ":XS
 5140   IF XS<=XI THEN 5070
 5150   INPUT AT 17,5,PROMPT "Unter
e y-Grenze: ":YI
 5160   INPUT AT 19,5,PROMPT "Obere
 y-Grenze:  ":YS
 5170   IF YS<=YI THEN 5070
 5180   CLOSE #101
 5190   TEXT 80
 5200   SET VIDEO MODE 1
 5210   SET VIDEO COLOUR 0
 5220   SET VIDEO X 40
 5230   SET VIDEO Y 25
 5240   OPEN #101:"video:"
 5250   DISPLAY #101:AT 1 FROM 1 TO
 25
 5260   CLEAR #101
```

```
 5270   SET #101:PALETTE BLACK,YELL
OW,RED,0,0,0,0,0
 5280   SET #101:LINE STYLE 5
 5290   LET MX=1280/(ABS(XI)+ABS(XS
))
 5300   LET MY=900/(ABS(YI)+ABS(YS)
)
 5310   LET UX=IP(ABS(XI)*MX)
 5320   LET UY=IP(ABS(YI)*MY)
 5330   FOR I=0 TO 13
 5340     LET X=I*92
 5350     PLOT #101:X,0;X,899
 5360     PLOT #101:X+4,70,
 5370     LET X=ROUND((X-UX)/MX,1)
 5380     PRINT #101:X;
 5390   NEXT I
 5400   FOR I=0 TO 9
 5410     LET Y=I*92
 5420     PLOT #101:0,Y;1279,Y
 5430     PLOT #101:8,Y+32,
 5440     LET Y=ROUND((Y-UY)/MY,1)
 5450     PRINT #101:Y;
 5460   NEXT I
 5470   SET #101:LINE STYLE 1
 5480 END DEF
 5490 !
 5500 !**** Pol.-Plotten ****
 5510 !
 5520 DEF F1_PLOT
 5530   LET TE=1
 5540   FOR I=XI TO XS STEP 5/MX
 5550     LET Y=C(N1)
 5560     FOR I2=N1-1 TO 1 STEP-1
 5570       LET Y=Y*I+C(I2)
 5580     NEXT I2
 5590     LET X=UX+I*MX:LET Y=UY+I*
MY
 5600     IF X<0 OR X>1279 OR Y<0 O
R Y>899 THEN 5620
 5610     PLOT #101:X,Y;
 5620   NEXT I
 5630 END DEF
 5640 !
 5650 !**** Display-Kopf ****
 5660 !
 5670 DEF HEAD
 5680   SET VIDEO MODE 1
 5690   SET VIDEO COLOUR 3
 5700   SET VIDEO X 40
 5710   SET VIDEO Y 5
 5720   OPEN #101:"video:"
 5730   CLEAR #101
 5740   DISPLAY #101:AT 1 FROM 1 TO
 5
 5750   PLOT #101:150,120,
 5760   PRINT #101:"Polynome"
 5770   PLOT #101:1,0;1279,0
 5780 END DEF
 5790 !
 5800 !**** Potentialfkt.-Plotten *
 ***
 5810 !
 5820 DEF F2_PLOT
 5830   FOR I=XI TO XS STEP 5/MX
 5840     LET Y=C(1)*I^C(2)
 5850     LET X=UX+I*MX:LET Y=UY+Y*
MY
 5860     IF X<0 OR X>1279 OR Y<0 O
R Y>899 THEN 5880
 5870     PLOT #101:X,Y;
 5880   NEXT I
 5890 END DEF
 5900 !
 5910 !**** Exponentialfkt.-Plotten
 ****
 5920 !
 5930 DEF F3_PLOT
 5940   FOR I=XI TO XS STEP 5/MX
 5950     LET Y=C(1)*EXP(C(2)*I)
 5960     LET X=UX+I*MX:LET Y=UY+Y*
MY
 5970     IF X<0 OR X>1279 OR Y<0 O
R Y>899 THEN 5990
 5980     PLOT #101:X,Y;
 5990   NEXT I
 6000 END DEF
 6010 !
 6020 !**** Vorgegebene Punkte einz
eichnen ****
 6030 !
 6040 DEF PUNKTE
 6050   SET BEAM OFF
 6060   FOR I=1 TO N2
 6070     LET X=UX+X2(I)*MX:LET Y=U
Y+Y2(I)*MY
 6080     IF X<0 OR X>1279 OR Y<0 O
R Y>899 THEN 6110
 6090     PLOT #101:X-8,Y+18,
 6100     PRINT #101:"x";
 6110   NEXT I
 6120 END DEF
```

Abschreibung

Rechner: Enterprise 64/128
Programmname: Abschreibungen
Programmlänge: ca. 10 500 Bytes
Programmiersprache: BASIC
Zubehör: Drucker optional

Die Zeit der Einkommensteuererklärungen naht, und mit ihr vergrößert sich die Zahl der blassen Gestalten die übernächtigt vor ihren Formularen sitzen. Sie können sich dies etwas angenehmer gestalten, wenn Sie mit Hilfe dieses Programms Ihre Abschreibungen errechnen können.

Programmablauf

Nach dem Starten erscheint das Menü. Allgemein ist zu sagen, daß Sie mit "m" in das jeweils höhere Menü gelangen. Im Hauptmenü bieten sich Ihnen folgende Möglichkeiten.

Lineare Abschreibung

Nun müssen Sie den Anschaffungswert des abzuschreibenden Objektes eingeben. Danach verlangt der Rechner noch die Nutzungsdauer (nur ganze Jahre!), in der abgeschrieben werden soll. Hierauf erfolgt eine Sicherheitsabfrage, die eine Bestätigung durch "j" erwartet (bei "n" Neueingabe der Daten). Nachdem Sie noch ein Ausgabegerät gewählt und das Afa-Jahr eingegeben haben, zeigt Ihnen das Programm die Grunddaten (weiter mit "v"). Das weitere Vorgehen siehe "Untermenü").

Degressive Abschreibung

Falls Ihnen der Prozentsatz nicht bekannt ist, errechnet diesen das Programm nach Ihren Angaben (Anschaffungswert,Restwert und Nutzungsdauer), andernfalls genügt die Eingabe des Anschaffungswerts und des Prozentsatzes. Auch hier erfolgt eine Sicherheitsabfrage, ob die gespeicherten Daten richtig sind. Die Ausgabe der Werte erfolgt analog zu "Lineare Abschreibung".

Digitale Abschreibung

Hier werden Sie nach Anschaffungswert sowie Nutzungsdauer gefragt; anschließend muß die Bestätigung der Daten erfolgen. Nachdem Sie das Anzeigen der Grunddaten mit "v" verlassen haben, verfahren Sie wie unter "Untermenü" beschrieben, allerdings mit einer Einschränkung: Das "Vorwärts-" bzw. "Rückwärtsblättern" ist wirkungslos, es wird stets ein Jahr darauf angezeigt.

Untermenü

Bei allen Abschreibungsarten wird auf ein Unterprogramm zurückgegriffen, das folgende Funktionen bietet:

* "d"
  Hiermit wird die Displayanzeige auf den Drucker ausgegeben.
* "m"
  Damit kommen Sie in das Menü der Ausgabegeräte.
* "r"
  Dies entspricht einem "Rückwärtsblättern", d.h. ein Jahr vorher wird angezeigt.
* "v"
  Das Gegenstück von "r".

Ich hoffe, daß Sie dieses Jahr mehr Geld vor der Steuer "retten" können!

Thomas Müller

BEIM ABTIPPEN ARBEIT SPAREN
können Sie sich, wenn Sie mit der Anforderungskarte am Heftende die Programme auf Datenträger anfordern.
Anwenden und auf Ihre Belange abändern lassen sich diese Programme allerdings nur mit Hilfe dieser CHIP-SPECIAL-Ausgabe.

```
Lineare Abschreibung
====================

Anschaffungswert:   8000
Nutzungsjahre:      4
Jährliche Abschr.:  25 %  =  1999.75


            1 . Jahr


Anschaffungswert:   8000
Abschreibung:    -1999.75
Restwert:         6000.25


            2 . Jahr


Restwert:         6000.25
Abschreibung:    -1999.75
Restwert:         4000.5


            3 . Jahr


Restwert:         4000.5
Abschreibung:    -1999.75
Restwert:         2000.75


            4 . Jahr


Restwert:         2000.75
Abschreibung:    -1999.75
Restwert:         1
```

```
100 !Abschreibungen
110 !für Chip Special
120 !von Thomas Müller
130 !
140 GOTO 180
150 CALL MENUE
160 END
170 !
180 !==== Initialisierung ====
190 !
200 SET STATUS OFF
210 STRING L$(1)*38,UP$(4)*35
220 DIM T(4)
230 LET L$(1),A$=""
240 LET AS,FF,DF,AW,RW,ND,Y,E,X=0
250 LET MA=1000000
260 FOR I=1 TO 32
270   LET L$(1)=L$(1)&CHR$(32)
280 NEXT I
290 FOR I=1 TO 4
300   READ UP$(I)
310   LET T(I)=19-IP(LEN(UP$(I))/
2)
320 NEXT I
330 !--- Datas für UP$ ---
340 DATA "Lineare Abschreibung"
350 DATA "Degressive Abschreibung
"
360 DATA "Digitale Abschreibung"
370 DATA "Programmende"
380 !---- window 1 ----
390 SET VIDEO MODE 1
400 SET VIDEO COLOUR 1
410 SET VIDEO X 40
420 SET VIDEO Y 25
430 OPEN #1:"video:"
440 SET #1:CURSOR CHARACTER 32
450 SET #1:CHARACTER 150,192,224,
240,248,252,248,240,224,192
460 SET #1:PALETTE BLACK,RGB(.6,.
6,.7),RED,BLUE,0,0,0,0
470 !--- Textparameter ---
480 SET #102:PALETTE BLACK,RGB(.6
,.6,.7),BLUE,RED,0,0,0,0
490 !---- window 2 ----
500 SET VIDEO MODE 1
510 SET VIDEO COLOUR 1
520 SET VIDEO X 40
530 SET VIDEO Y 4
540 OPEN #2:"video:"
550 SET #2:CURSOR CHARACTER 32
560 SET #2:PALETTE BLACK,RGB(.6,.
6,.7),RED,BLUE,0,0,0,0
570 !---- window 3 ----
580 SET VIDEO MODE 5
590 SET VIDEO COLOUR 0
600 SET VIDEO X 40
610 SET VIDEO Y 25
620 OPEN #3:"video:"
630 SET #3:CURSOR CHARACTER 32
640 GOTO 150
650 !
660 !==== Menü ====
670 !
680 DEF MENUE
690   DISPLAY #1:AT 1 FROM 1 TO 2
5
700   SET #1:INK 1
710   PLOT #1:550,800,
720   PRINT #1:"Menü"
730   PLOT #1:500,850;720,850;720
,720;500,720;500,850,512,830,
```

```
  740    SET #1:INK 3
  750    PLOT #1:PAINT,
  760    FOR I=1 TO 4
  770      PLOT #1:T(I)*32,720-I*72,
  780      PRINT #1:UP$(I);
  790    NEXT I
  800    !--- Auswahl ---
  810    LET I,G=1
  820    SET #1:INK 0
  830    PLOT #1:T(G)*32-32,720-G*72
,
  840    PRINT #1:CHR$(159);
  850    SET #1:INK 1
  860    PLOT #1:T(I)*32-32,720-I*72
,
  870    PRINT #1:CHR$(150);
  880    CALL WAIT
  890    LET AS=ORD(A$)
  900    IF AS=176 THEN LET G=I:LET
I=I-SGN(I-1):GOTO 820
  910    IF AS=180 THEN LET G=I:LET
I=I+SGN(4-I):GOTO 820
  920    IF AS<>32 THEN 880
  930    LET AS=I
  940    SET #1:INK 0
  950    PLOT #1:T(I)*32-32,720-I*72
,
  960    PRINT #1:CHR$(159);
  970    DISPLAY #3:AT 1 FROM 1 TO 2
5
  980    !--- Aufrufe ---
  990    SELECT CASE AS
 1000      !--- linear ---
 1010    CASE 1
 1020      CALL LINEAR
 1030      !--- degressiv ---
 1040    CASE 2
 1050      CALL DEGRESSIV
 1060      !--- digital ---
 1070    CASE 3
 1080      CALL DIGITAL
 1090      !--- Ende ---
 1100    CASE 4
 1110      CALL PRGENDE
 1120      GOTO 1150
 1130    END SELECT
 1140    GOTO 690
 1150  END DEF
 1160  !
 1170  !==== Lineare Abschr. ====
 1180  !
 1190  DEF LINEAR
 1200    LET Z=1:LET AW=0:LET ND=0
 1210    CALL DESK1
 1220    CLEAR TEXT
 1230    DISPLAY #102:AT 5 FROM 5 TO
 24
 1240    PRINT AT 6,0:
 1250    PRINT
 1260    CALL AW_
 1270    IF AW=0 THEN 1740
 1280    IF AW>MA THEN PRINT CHR$(17
6);CHR$(176);:GOTO 1260
 1290    CALL ND_
 1300    LET ND=ABS(IP(ND))
 1310    IF ND=0 THEN PRINT CHR$(176
);CHR$(176);:GOTO 1290
 1320    CALL SURE
 1330    IF FF=1 THEN 1200
 1340    !--- Berechnung ---
 1350    LET AB=(AW-1)/ND
 1360    LET P=AB/AW*100
 1370    !--- Ausgabe ---
 1380    CALL DEVICE
 1390    IF DF=2 THEN 1740
 1400    CALL AFA1
 1410    IF Y=0 THEN LET Y=1
 1420    SELECT CASE DF
 1430      !--- Bildschirm ---
 1440    CASE 0
 1450      CLEAR TEXT
 1460      PRINT AT 3,5:UP$(1)
 1470      PRINT AT 6,0:"Nutzungsjah
re:      ";ND
 1480      PRINT AT 8,0:"Jährliche A
bschr.: ";ROUND(P,0);"%"
 1490      CALL MENUE2
 1500      IF A$="m" THEN 1380
 1510      LET X=Y
 1520      LET J=X:LET RA=AW-AB*(X-1
):LET RN=RA-AB
 1530      CALL AUSG2
 1540      CALL MENUE2
 1550      IF A$="m" THEN 1380
 1560      GOTO 1520
 1570      !--- Drucker ---
 1580    CASE 1
 1590      CALL AFA2
 1600      IF E=0 THEN LET E=ND
 1610      LPRINT "Lineare Abschreib
ung"
 1620      LPRINT "==================
==="
 1630      LPRINT
 1640      LPRINT "Anschaffungswert:
 ";AW
 1650      LPRINT "Nutzungsjahre:
 ";ND
 1660      LPRINT "Jährliche Abschr.
: ";ROUND(P,0);"%  = ";AB
 1670      LPRINT
 1680      FOR X=Y TO E
 1690        LET J=X:LET RA=AW-(X-1)
```

```
*AB:LET RN=RA-AB
 1700       CALL AUSG3
 1710     NEXT X
 1720     GOTO 1380
 1730   END SELECT
 1740 END DEF
 1750 !
 1760 !==== Degressive Abschreibung
 ====
 1770 !
 1780 DEF DEGRESSIV
 1790   LET P,AW=0:LET Z=1:LET ND=M
A
 1800   CALL DESK1
 1810   CLEAR TEXT
 1820   DISPLAY #102:AT 5 FROM 5 TO
 24
 1830   PRINT AT 7,5:"Ist Ihnen der
 Prozentsatz"
 1840   PRINT AT 9,10:"bekannt (j/n
)?"
 1850   CALL WAIT
 1860   IF A$="j" THEN CALL PROZENT
:GOTO 2370
 1870   CLEAR TEXT
 1880   PRINT AT 6,0:
 1890   CALL AW_
 1900   IF AW=0 THEN 2370
 1910   IF AW>MA THEN PRINT CHR$(17
6);CHR$(176);:GOTO 1890
 1920   CALL RW_
 1930   IF RW<=0 THEN PRINT CHR$(17
6);CHR$(176);:GOTO 1920
 1940   CALL ND_
 1950   IF ND<=0 THEN PRINT CHR$(17
6);CHR$(176);:GOTO 1940
 1960   CALL SURE
 1970   IF FF=1 THEN 1790
 1980   CALL DEVICE
 1990   IF DF=2 THEN 2370
 2000   LET P=100*(1-10^(LOG10(RW/A
W)/ND)):LET R=1-(P/100)
 2010   CALL AFA1
 2020   LET Y=ABS(IP(Y))
 2030   IF Y>ND THEN PRINT CHR$(176
);CHR$(176);:GOTO 2010
 2040   SELECT CASE DF
 2050     !--- Bildschirm ---
 2060   CASE 0
 2070     LET N1=ND:LET ND=MA
 2080     CALL AFA1
 2090     LET ND=N1
 2100     CLEAR TEXT
 2110     PRINT AT 6,0:"Anschaffung
swert: ";AW
 2120     PRINT AT 8,0:"Restwert:
      ";RW
 2130     PRINT AT 10,0:"Afa - Jahr
e:     ";N1
 2140     PRINT AT 12,0:"Prozentsat
z:     ";P
 2150     CALL MENUE2
 2160     IF A$="m" THEN 1980
 2170     LET X=Y-1
 2180     IF X=-1 THEN LET X=0
 2190     LET J=X:LET RA=AW*R^(X-1)
:LET AB=RA*P/100:LET RN=RA-AB
 2200     CALL AUSG2
 2210     CALL MENUE2
 2220     IF A$="m" THEN 1980
 2230     GOTO 2190
 2240     !--- Drucker ---
 2250   CASE 1
 2260     CALL AFA2
 2270     IF Y=0 THEN LET Y=1
 2280     IF E=0 THEN 1980
 2290     LET X=Y-1
 2300     CALL DEGAUSG
 2310     LET X=X+1:LET J=X:LET RA=
AW*R^(X-1):LET AB=RA*P/100:LET RN=R
A-AB
 2320     CALL AUSG3
 2330     IF X=E THEN 2350
 2340     GOTO 2310
 2350     GOTO 1980
 2360   END SELECT
 2370 END DEF
 2380 !
 2390 !==== Digitale Abschreibung =
===
 2400 !
 2410 DEF DIGITAL
 2420   LET SU,AW,ND=0
 2430   CALL DESK1
 2440   CLEAR TEXT
 2450   DISPLAY #102:AT 5 FROM 5 TO
 24
 2460   PRINT AT 6,0:
 2470   PRINT
 2480   CALL AW_
 2490   IF AW=0 THEN 3130
 2500   IF AW>MA THEN PRINT CHR$(17
6);CHR$(176);:GOTO 2480
 2510   CALL ND_
 2520   LET ND=ABS(IP(ND))
 2530   IF ND=0 THEN PRINT CHR$(176
);CHR$(176);:GOTO 2510
 2540   CALL SURE
 2550   IF FF=1 THEN 2440
 2560   FOR X=1 TO ND
 2570     LET SU=SU+X
 2580   NEXT X
 2590   LET Q=(AW-1)/SU:LET Z=ND
 2600   LET AC=0
```

```
 2610    CALL DEVICE
 2620    IF DF=2 THEN 3130
 2630    SELECT CASE DF
 2640      !--- Bildschirm ---
 2650    CASE 0
 2660      CLEAR TEXT
 2670      PRINT AT 6,0:
 2680      CALL AFA1
 2690      LET Y=ABS(IP(Y))
 2700      IF Y=0 OR Y=1 THEN LET Y=
1:GOTO 2740
 2710      FOR X=ND TO ND-Y+2 STEP-1

 2720        LET AB=X*Q:LET AC=AC+AB

 2730      NEXT X
 2740      CLEAR TEXT
 2750      PRINT AT 6,0:"Anschaffung
swert: ";AW
 2760      PRINT AT 8,0:"Afa - Jahre
:      ";ND
 2770      CALL MENUE2
 2780      IF A$="m" THEN 2600
 2790      FOR X=ND-Y+1 TO 1 STEP-1
 2800        LET AB=X*Q:LET RN=IP(10
0*(AW-AB-AC)+.5)/100:LET RA=AW-AC:L
ET J=ND-X+1
 2810        CALL AUSG2
 2820        LET AC=AC+AB
 2830        CALL MENUE2
 2840        IF A$="m" THEN EXIT FOR

 2850        IF A$="r" THEN LET X=X+
SGN(ND-Y+1-X)
 2860        IF A$="v" AND J=1 THEN
2880
 2870        IF A$="v" THEN LET X=X-
SGN(X-1)
 2880      NEXT X
 2890      GOTO 2600
 2900      !--- Drucker ---
 2910    CASE 1
 2920      LET AC=0
 2930      CALL AFA1
 2940      CALL AFA2
 2950      LET Y=ABS(IP(Y)):LET E=AB
S(IP(E))
 2960      IF E=0 THEN LET E=1
 2970      IF Y=0 OR Y=1 THEN LET Y=
1:GOTO 3010
 2980      FOR X=N TO N-Y+2 STEP-1
 2990        LET AB=X*Q:LET AC=AC+AB

 3000      NEXT X
 3010      LPRINT "Digitale Abschrei
bung"
 3020      LPRINT "================
====="
 3030      LPRINT :LPRINT
 3040      LPRINT "Anschaffungswert:
";AW
 3050      LPRINT "Afa - Jahre:
";ND
 3060      FOR X=ND-Y+1 TO ND+1-E ST
EP-1
 3070        LET AB=X*Q:LET RN=AW-AB
-AC:LET RA=AW-AC:LET J=ND-X+1
 3080        CALL AUSG3
 3090        LET AC=AC+AB
 3100      NEXT X
 3110      GOTO 2600
 3120    END SELECT
 3130 END DEF
 3140 !
 3150 !==== Programmende ====
 3160 !
 3170 DEF PRGENDE
 3180    CLOSE #2
 3190    CLOSE #1
 3200    TEXT 40
 3210 END DEF
 3220 !
 3230 !==== Druckerausgabe ====
 3240 !
 3250 DEF AUSG3
 3260    LPRINT
 3270    LPRINT
 3280    LPRINT "            ";J;". J
ahr"
 3290    LPRINT :LPRINT
 3300    IF X=Z THEN LPRINT "Anschaf
fungswert: ";AW:GOTO 3320
 3310    LPRINT "Restwert:       ";RA

 3320    LPRINT "Abschreibung:   ";-A
B
 3330    LPRINT "Restwert:       ";RN

 3340 END DEF
 3350 !
 3360 !==== Degr. Abschr. Ausdruck
====
 3370 !
 3380 DEF DEGAUSG
 3390    LPRINT "Degressive Abschrei
bung"
 3400    LPRINT "================
====="
 3410    LPRINT :LPRINT
 3420    LPRINT "Anschaffungswert: "
;AW
 3430    LPRINT "Restwert:         "
;RW
 3440    LPRINT "Nutzungsjahre:    "
```

```
;ND
 3450   LPRINT "Prozentsatz:      "
;P
 3460   LPRINT :LPRINT
 3470 END DEF
 3480 !
 3490 !==== Eingabeüberschrift ====
 3500 !
 3510 DEF DESK1
 3520   CLEAR #2
 3530   SET #2:INK 1
 3540   PLOT #2:500,100,
 3550   PRINT #2:"Eingabe"
 3560   PLOT #2:450,140;790,140;790
,20;450,20;450,140
 3570   SET #2:INK 3
 3580   PLOT #2:455,130,PAINT,
 3590   DISPLAY #2:AT 1 FROM 1 TO 4
 3600 END DEF
 3610 !
 3620 !==== Sicherheitsabfrage ====
 3630 !
 3640 DEF SURE
 3650   PRINT :PRINT "Sind die Date
n korrekt? (j/n)"
 3660   CALL WAIT
 3670   IF A$="j" THEN LET FF=0:GOT
O 3700
 3680   IF A$<>"n" THEN 3660
 3690   LET FF=1
 3700 END DEF
 3710 !
 3720 !==== Ausgabegerät ====
 3730 !
 3740 DEF DEVICE
 3750   CLEAR TEXT
 3760   DISPLAY #102:AT 1 FROM 1 TO
 24
 3770   PRINT AT 5,10:"Ausgabegerät
:"
 3780   PRINT AT 6,10:"============
="
 3790   PRINT AT 9,10:"1 <= Bildsch
irm"
 3800   PRINT AT 11,10:"2 <= Drucke
r"
 3810   PRINT AT 13,10:"3 <= Menü"
 3820   CALL WAIT
 3830   LET DF=VAL(A$)-1
 3840   IF DF<0 OR DF>2 THEN 3830
 3850   CLEAR TEXT
 3860 END DEF
 3870 !
 3880 !==== Untermenü ====
 3890 !
 3900 DEF MENUE2
 3910   CALL WAIT
 3920   IF A$="m" THEN 3970
 3930   IF A$="r" THEN LET X=X-SGN(
X-Y):GOTO 3970
 3940   IF A$="v" THEN LET X=X+SGN(
ND-X):GOTO 3970
 3950   IF A$<>"d" THEN 3910
 3960   COPY
 3970 END DEF
 3980 !
 3990 !==== Zwischenausgabe ====
 4000 !
 4010 DEF AUSG2
 4020   CLEAR TEXT
 4030   PRINT AT 3,10:J;". Jahr"
 4040   PRINT AT 6,0:
 4050   PRINT
 4060   IF X=Z THEN PRINT "Anschaff
ungswert: ";AW:GOTO 4080
 4070   PRINT :PRINT "Restwert:
    ";RA
 4080   PRINT :PRINT "Abschreibung:
    ";-AB
 4090   PRINT :PRINT "Restwert:
    ";RN
 4100 END DEF
 4110 !
 4120 !==== Prozent eingegeben ====
 4130 !
 4140 DEF PROZENT
 4150   LET ND=MA
 4160   CLEAR TEXT
 4170   PRINT AT 6,0:
 4180   CALL AW_
 4190   IF AW=0 THEN 4610
 4200   IF AW>MA THEN PRINT CHR$(17
6);CHR$(176);:GOTO 4180
 4210   PRINT :INPUT PROMPT "Jährl.
 Abschr. in %: ":P
 4220   IF P=0 THEN 4610
 4230   CALL SURE
 4240   IF FF=1 THEN 4160
 4250   LET X=0:LET R=1-(P/100)
 4260   CALL DEVICE
 4270   IF DF=2 THEN 4610
 4280   SELECT CASE DF
 4290     !--- Bildschirm ---
 4300   CASE 0
 4310     LET ND=MA
 4320     CALL AFA1
 4330     CLEAR TEXT
 4340     PRINT AT 6,0:"Anschaffung
swert: ";AW
 4350     PRINT AT 8,0:"Prozentsatz
```

```
:      ";P
 4360      CALL MENUE2
 4370      IF A$="m" THEN 4260
 4380      LET X=Y-1
 4390      IF X=-1 THEN LET X=0
 4400      LET X=X+1
 4410      LET J=X:LET RA=AW*R^(X-1)
:LET AB=RA*P/100:LET RN=RA-AB
 4420      CALL AUSG2
 4430      CALL MENUE2
 4440      IF A$="m" THEN 4610
 4450      GOTO 4410
 4460      !--- Drucker ---
 4470    CASE 1
 4480      CALL AFA1
 4490      LET Y=ABS(IP(Y))
 4500      CALL AFA2
 4510      LET E=ABS(IP(E))
 4520      IF Y=0 THEN LET Y=1
 4530      IF E=0 THEN 3890
 4540      LET X=Y-1
 4550      CALL DEGAUSG
 4560      LET X=X+1:LET J=X:LET RA=
AW*R^(X-1):LET AB=RA*P/100:LET RN=R
A-AB
 4570      CALL AUSG3
 4580      IF X=E THEN 4610
 4590      GOTO 4560
 4600    END SELECT
 4610 END DEF
 4620 !
 4630 !==== Tastendruck ====
 4640 !
 4650 DEF WAIT
 4660   LET A$=INKEY$
 4670   IF A$="" THEN 4660
 4680 END DEF
 4690 !
 4700 !==== Eingaben ====
 4710 !
 4720 DEF AW_
 4730   PRINT :INPUT PROMPT "Anscha
ffungswert: ":AW
 4740 END DEF
 4750 DEF ND_
 4760   PRINT :INPUT PROMPT "Nutzun
gsdauer:    ":ND
 4770 END DEF
 4780 DEF AFA1
 4790   INPUT AT 5,0,PROMPT "Ausgab
e ab Afa-Jahr: ":Y
 4800   LET Y=ABS(IP(Y))
 4810   IF Y>ND THEN 4790
 4820 END DEF
 4830 DEF RW_
 4840   PRINT :INPUT PROMPT "Restwe
rt (> 0!!!): ":RW
 4850 END DEF
 4860 DEF AFA2
 4870   INPUT AT 7,0,PROMPT "Ausgab
e bis Afa-Jahr: ":E
 4880 END DEF
```

Teilchenbahnen

Rechner: Enterprise 64/128
Programmname: Teilchenbahnen
Programmlänge: ca. 10 500 Bytes
Programmiersprache: BASIC
Zubehör: Drucker optional

Wer unter den jüngeren Lesern hat im Physikunterricht nicht auch schon jene leidvolle Erfahrung mit einem großen Teilgebiet der Physik gemacht: Bahnen geladener Teilchen in elektrischen und magnetischen Feldern. Alljährlich steht dieses Thema u.a. auch in den Abituraufgaben zur Debatte. Damit dieses komplexe Gebiet anschaulich erschlossen werden kann, wurde dieses Programm geschrieben.
Zwar ist es nicht möglich, alle Aspekte abzudecken, jedoch die gebräuchlichsten Aufgabenstellungen werden hierdurch nähergebracht.

Das Menü

Nach dem Starten wählen Sie im Menü mit dem Cursorstick (nach oben bzw. unten) den gewünschten Versuchsaufbau an und drücken <SPACE>.

Wienfilter

Das Wienfilter besteht aus einem Plattenkondensator, dessen elektrisches Feld mit einem im Lot dazu (vom Betrachter weg) stehenden Magnetfeld gekreuzt ist. Also benötigt das Programm hier den Plattenabstand; daraus errechnet es einen Darstellungsfaktor, bei dem der

Plattenkondensator maßstabsgetreu abgebildet wäre. Sie können allerdings natürlich auch einen anderen Faktor wählen, z.B. um die Bahn genauer verfolgen zu können. Als nächstes müssen Sie die Kondensatorspannung, die Stärke des B-Feldes, die elektrische Ladung des Teilchens sowie dessen Masse und die Geschwindgkeit in x-Richtung eingeben.
Während der Simulation können Sie jederzeit durch Drücken von "s" in die Ausgabe springen. Ansonsten läuft die Simulation, bis das Teilchen eine Platte erreicht.
Nun stehen sowohl der Drucker als auch das Display zur Ausgabe zur Verfügung. Dabei werden zuerst die Anfangsdaten gezeigt und nach einem Tastendruck die Enddaten. Nach einem weiteren Tippen verzweigt das Programm ins Menü.

Massenspektrograph

Der Massenspektrograph besteht aus einer Blende und einem parallel dazu verlaufenden Magnetfeld. Diesem Aufbau ist ein Wienfilter als Geschwindigkeitsfilter vorgeschaltet, so daß man durch Messung des Radius der Bahn, die die Teilchen durchfliegen, die Masse der Teilchen bestimmen kann.
Hierfür werden die Stärke des B-Feldes, die Ladung und die Masse (der Rechner kann ja keine Teilchen fliegen lassen!) des Teilchens und die Geschwindigkeit in x-Richtung benötigt. Auch hier kann die Simulation mit "s" gestoppt werden. Der übrige Ablauf ist analog zu "Wienfilter".

Schraubenbahn I

Bewegt sich ein geladenes Teilchen durch ein Magnetfeld, so durchläuft es eine Schraubenbahn (Ausnahme: es fliegt genau parallel). Diese Situation wird hier simuliert: Dazu müssen Sie die Geschwindigkeit, den Winkel zum B-Feld, die Stärke des B-Feldes sowie die Ladung und die Masse des Teilchens eingeben. Nun durchläuft dieses Teilchen eine Periode seiner Schraubenbahn, wobei das Verhältnis von Radius zu Periodenlänge deutlich werden soll.
Anschließend erfolgt nach einem Tastendruck die Ausgabe der Werte; Nach erneutem Tippen erscheint wieder das Menü.

Schraubenbahn II

Dieses Unterprogramm soll eine dreidimensionale Ansicht einer derartigen Schraubenbahn vermitteln. Dabei werden nachfolgende Daten berücksichtigt: Die Geschwindigkeit, der Winkel zum B-Feld sowie der Betrachtungswinkel, die Stärke des B-Feldes, die Ladung und Masse des Teilchens und den Darstellungsfaktor.

Programmende

Damit werden alle Fenster geschlossen; zudem wird wieder in den 40-Zeichen-Modus geschaltet.

Thomas Müller

```
100 !Teilchenbahnen
110 !für Chip Special
120 !von Thomas Müller
130 !
140 GOTO 180
150 CALL MENUE
160 END
170 !
180 !**** Initialisierung ****
190 !
200 TEXT
210 CLEAR TEXT:SET STATUS OFF
220 DIM T(5)
230 STRING UP$(5)*30,L$(1)*70
240 LET A$,L$(1)=""
250 LET D,FA,E,UK,B,Q,M,TI,VX,VY,
X,Y,V1,F,AS=0
260 LET MX,MY,R,R2,VF,H,G1,G2,G3,
G4,DF,V,WA,BW,S=0
270 FOR I=1 TO 70
280   LET L$(1)=L$(1)&CHR$(159)
290 NEXT I
300 FOR I=1 TO 5
310   READ UP$(I)
320   LET T(I)=9-(LEN(UP$(I))/2)
330 NEXT I
```

```
  340 DATA "Wienfilter"
  350 DATA "Massenspektrograph"
  360 DATA "Schraubenbahn I"
  370 DATA "Schraubenbahn II"
  380 DATA "Programmende"
  390 !--- window 1 ---
  400 SET VIDEO MODE 1
  410 SET VIDEO COLOUR 2
  420 SET VIDEO X 42
  430 SET VIDEO Y 3
  440 OPEN #1:"video:"
  450 !--- window 2 ---
  460 SET VIDEO MODE 0
  470 SET VIDEO COLOUR 0
  480 SET VIDEO X 42
  490 SET VIDEO Y 25
  500 OPEN #5:"video:"
  510 SET #5:CURSOR CHARACTER 32
  520 !--- textparameter ---
  530 SET #102:PALETTE BLACK,RGB(.6
,.6,.7),BLUE,RED,0,0,0,0
  540 !--- window 3 ---
  550 SET VIDEO MODE 1
  560 SET VIDEO COLOUR 1
  570 SET VIDEO X 42
  580 SET VIDEO Y 25
  590 OPEN #101:"video:"
  600 CLEAR #101
  610 SET #101:PALETTE BLACK,BLUE,R
GB(.6,.6,.7),RED,0,0,0,0
  620 SET #101:INK 3
  630 FOR I=0 TO 1340 STEP 100
  640   FOR I2=0 TO 890 STEP 100
  650     PLOT #101:I,I2,
  660     PRINT #101:"x";
  670   NEXT I2
  680 NEXT I
  690 SET #101:INK 2
  700 PLOT #101:550,550,
  710 PRINT #101:"Teilchen-";
  720 PLOT #101:600,450,
  730 PRINT #101:"bahnen";
  740 DISPLAY #101:AT 1 FROM 1 TO 2
5
  750 FOR I=0 TO 12*PI STEP PI/90
  760   SET #101:INK 2
  770   PLOT #101:672+COS(I)*300,45
0+SIN(I)*300,ELLIPSE 12,12,
  780   LET A$=INKEY$
  790   IF A$<>"" THEN EXIT FOR
  800   SET #101:INK 0
  810   PLOT #101:ELLIPSE 12,12,
  820 NEXT I
  830 CLOSE #101
  840 !--- window 4 ---
  850 SET VIDEO MODE 1
  860 SET VIDEO COLOUR 2
  870 SET VIDEO X 42
  880 SET VIDEO Y 22
  890 OPEN #2:"video:"
  900 SET #2:PALETTE BLACK,WHITE,BL
UE,YELLOW,RED,CYAN,0,0
  910 SET #2:CHARACTER 150,192,224,
240,248,252,248,240,224,192
  920 !--- window 5 ---
  930 SET VIDEO MODE 1
  940 SET VIDEO COLOUR 0
  950 SET VIDEO X 42
  960 SET VIDEO Y 25
  970 OPEN #4:"video:"
  980 SET #4:PALETTE BLACK,RGB(.6,.
6,.7),BLUE,RED,0,0,0,0
  990 GOTO 150
 1000 !
 1010 !**** Menü ****
 1020 !
 1030 DEF MENUE
 1040   DISPLAY #5:AT 1 FROM 1 TO 2
5
 1050   PLOT #1:250,100,
 1060   SET #1:INK 2
 1070   PRINT #1:"Teilchenbahnen"
 1080   PLOT #1:0,8;1343,8
 1090   FOR I=1 TO 5
 1100     PLOT #2:100+T(I)*64,750-I
*100,
 1110     SET #2:INK I
 1120     PRINT #2:UP$(I)
 1130   NEXT I
 1140   DISPLAY #2:AT 4 FROM 1 TO 2
2
 1150   DISPLAY #1:AT 1 FROM 1 TO 3

 1160   !--- Auswahl ---
 1170   LET I,G=1
 1180   SET #2:INK 0
 1190   PLOT #2:T(G)*64+20,758-G*10
0,
 1200   PRINT #2:CHR$(159);
 1210   SET #2:INK 5
 1220   PLOT #2:T(I)*64+20,758-I*10
0,
 1230   PRINT #2:CHR$(150);
 1240   CALL WAIT
 1250   LET AS=ORD(A$)
 1260   IF AS=176 THEN LET G=I:LET
I=I-SGN(I-1):GOTO 1180
 1270   IF AS=180 THEN LET G=I:LET
I=I+SGN(5-I):GOTO 1180
 1280   IF AS<>32 THEN 1240
 1290   LET AS=I
 1300   SET #2:INK 0
 1310   PLOT #2:T(I)*64+20,758-I*10
0,
```

```
 1320   PRINT #2:CHR$(159);
 1330   !--- Aufrufe ---
 1340   SELECT CASE AS
 1350     !--- Wienfilter ---
 1360   CASE 1
 1370     CALL EINGABE1
 1380     CALL FILTER
 1390     CALL AUSGABE1
 1400     !--- Massenspektrograph -
--
 1410   CASE 2
 1420     CALL EINGABE2
 1430     CALL MSG
 1440     CALL AUSGABE2
 1450     !--- Schraubenbahn I ---
 1460   CASE 3
 1470     CALL EINGABE3
 1480     CALL SBI
 1490     CALL AUSGABE3
 1500     !--- Schraubenbahn II ---

 1510   CASE 4
 1520     CALL EINGABE4
 1530     CALL SBII
 1540     !--- Programmende ---
 1550   CASE 5
 1560     CLOSE #1
 1570     CLOSE #2
 1580     CLOSE #5
 1590     TEXT 40
 1600     GOTO 1640
 1610   CASE ELSE
 1620   END SELECT
 1630   GOTO 1140
 1640 END DEF
 1650 !
 1660 !**** Eingaben Wienfilter ***
*
 1670 !
 1680 DEF EINGABE1
 1690   CALL DESK1
 1700   CALL D_
 1710   CALL FA_
 1720   CALL E_
 1730   CALL B_
 1740   CALL Q_
 1750   CALL M_
 1760   CALL VX_
 1770 END DEF
 1780 !
 1790 !**** Wienfilter ****
 1800 !
 1810 DEF FILTER
 1820   !--- Init ---
 1830   LET VY=0:LET Y2=450:LET X2=
20:LET V1=VX:LET X,Y=0
 1840   !--- Bildschirmaufbau ---
 1850   CLEAR #4
 1860   DISPLAY #4:AT 1 FROM 1 TO 2
5
 1870   PLOT #4:20,0;1330,0;1330,50
;20,50;20,0
 1880   PLOT #4:20,899;1330,899;133
0,850;20,850;20,899
 1890   LET G3=90:LET G4=860:LET G1
=20:LET G2=1320
 1900   CALL BPLOT
 1910   !--- Beginn Simulation ---
 1920   LET TI=1/(ABS(VY*FA)+ABS(VX
*FA))
 1930   LET LY=Q*VX*B:LET LX=-Q*VY*
B
 1940   LET EY=Q*E
 1950   LET FY=LY-EY:LET FX=LX
 1960   IF (ABS(FY*1000000000)<EY)
AND(ABS(Y*1000000000)<LY) THEN LET
FY=0
 1970   LET AY=FY/M:LET AX=FX/M
 1980   LET Y=Y+VY*TI+AY/2*TI*TI:LE
T X=X+VX*TI+AX/2*TI*TI
 1990   LET VY=VY+AY*TI:LET VX=VX+A
X*TI
 2000   IF (X*FA<0) OR(X*FA>1310) O
R(Y*FA+Y2<50) OR(Y*FA+Y2>850) THEN
2110
 2010   SET #4:INK 0
 2020   PLOT #4:50,890,
 2030   PRINT #4:L$(1);
 2040   SET #4:INK 1
 2050   PLOT #4:50,890,
 2060   PRINT #4:"vx = ";VX;"   vy
= ";VY;
 2070   PLOT #4:X*FA+X2,Y*FA+Y2
 2080   LET A$=INKEY$
 2090   IF A$<>"s" THEN 1920
 2100   GOTO 2150
 2110   !--- "Weiterverfolgen" ---
 2120   IF X*FA<=1310 THEN 2150
 2130   LET X=0
 2140   GOTO 1860
 2150   !--- Ende der Simulation --
-
 2160   PING
 2170   PING
 2180   CALL WAIT
 2190 END DEF
 2200 !
 2210 !**** Ausgabe Wienfilter ****

 2220 !
 2230 DEF AUSGABE1
 2240   CLEAR TEXT
 2250   DISPLAY #5:AT 25 FROM 1 TO
1
```

```
 2260   DISPLAY #102:AT 1 FROM 1 TO
 24
 2270   CALL DRUCK
 2280   CLEAR TEXT
 2290   PRINT AT 3,10:"Anfangsdaten
"
 2300   PRINT AT 4,10:"===========
"
 2310   PRINT AT 6,8:"vx: ";V1
 2320   PRINT AT 7,8:"vy: ";0
 2330   PRINT AT 8,9:"x: ";0
 2340   PRINT AT 9,9:"y: ";0
 2350   PRINT AT 10,9:"E: ";E
 2360   PRINT AT 11,9:"B: ";B
 2370   PRINT AT 12,9:"q: ";Q
 2380   PRINT AT 13,9:"m: ";M
 2390   PRINT AT 14,4:"Faktor: ";FA

 2400   CALL WAIT
 2410   IF DF=1 THEN COPY
 2420   CLEAR TEXT
 2430   PRINT AT 3,10:"Enddaten"
 2440   PRINT AT 4,10:"========"
 2450   PRINT AT 6,8:"vx: ";VX
 2460   PRINT AT 7,8:"vy: ";VY
 2470   PRINT AT 8,9:"x: ";X
 2480   PRINT AT 9,9:"y: ";Y
 2490   CALL WAIT
 2500   IF DF=1 THEN COPY
 2510 END DEF
 2520 !
 2530 !**** Eingabe MSG ****
 2540 !
 2550 DEF EINGABE2
 2560   CALL DESK1
 2570   CALL B_
 2580   CALL Q_
 2590   CALL M_
 2600   CALL VX_
 2610 END DEF
 2620 !
 2630 !**** MSG ****
 2640 !
 2650 DEF MSG
 2660   !--- Initialisierung ---
 2670   LET MX=500
 2680   LET R=(M*VX)/(Q*B):LET VF=S
GN(R)
 2690   LET R=ABS(R)
 2700   LET R2=IP(R/(10^IP(LOG10(R)
))*20)
 2710   IF R2<20 THEN LET R2=R2*10
 2720   LET MY=450+R2*VF
 2730   !--- Bildschirmaufbau ---
 2740   CLEAR #4
 2750   DISPLAY #4:AT 1 FROM 1 TO 2
5
 2760   PLOT #4:500,0;900,0;900,50;
500,50;500,0
 2770   PLOT #4:500,899;900,899;900
,850;500,850;500,899
 2780   PLOT #4:500,50;500,440,500,
460;500,850
 2790   LET G1=500:LET G2=900:LET G
3=100:LET G4=840
 2800   CALL BPLOT
 2810   !--- Beginn Simulation ---
 2820   FOR I=-PI/2*SGN(Q) TO PI/2*
SGN(Q) STEP PI/180*SGN(Q)
 2830     LET X=MX+COS(I)*R2:LET Y=
MY+SIN(I)*R2
 2840     PLOT #4:X,Y
 2850     LET A$=INKEY$
 2860     IF A$="s" THEN EXIT FOR
 2870   NEXT I
 2880   PING
 2890   PING
 2900 END DEF
 2910 !
 2920 !**** Ausgabe MSG ****
 2930 !
 2940 DEF AUSGABE2
 2950   CLEAR TEXT
 2960   DISPLAY #5:AT 25 FROM 1 TO
1
 2970   DISPLAY #102:AT 1 FROM 1 TO
 24
 2980   CALL DRUCK
 2990   CLEAR TEXT
 3000   PRINT AT 3,10:"Anfangsdaten
"
 3010   PRINT AT 4,10:"===========
"
 3020   PRINT AT 6,8:"vx: ";VX
 3030   PRINT AT 7,8:"vy: ";0
 3040   PRINT AT 8,9:"x: ";0
 3050   PRINT AT 9,9:"y: ";0
 3060   PRINT AT 10,9:"B: ";B
 3070   PRINT AT 11,9:"q: ";Q
 3080   PRINT AT 12,9:"m: ";M
 3090   CALL WAIT
 3100   IF DF=1 THEN COPY
 3110   CLEAR TEXT
 3120   PRINT AT 3,10:"Enddaten"
 3130   PRINT AT 4,10:"========"
 3140   PRINT AT 6,8:"vx: ";-VX
 3150   PRINT AT 7,8:"vy: ";0
 3160   PRINT AT 8,9:"x: ";0
 3170   PRINT AT 9,9:"y: ";R*VF
 3180   CALL WAIT
 3190   IF DF=1 THEN COPY
 3200 END DEF
 3210 !
 3220 !**** Eingabe Schraubenb. I *
```

```
***
 3230 !
 3240 DEF EINGABE3
 3250   CALL DESK1
 3260   CALL V_
 3270   CALL WA_
 3280   CALL B_
 3290   CALL Q_
 3300   CALL M_
 3310 END DEF
 3320 !
 3330 !**** Schraubenb. I ****
 3340 !
 3350 DEF SBI
 3360   !--- Initialisierung ---
 3370   LET R=(M*SIN(WA)*V)/(B*Q)
 3380   LET VX=COS(WA)*V
 3390   LET D=(VX*2*PI*M)/(B*Q)
 3400   LET R=ABS(R)
 3410   IF D=0 THEN LET S=0:LET R2=
400:GOTO 3430
 3420   LET S=500/PI:LET R2=R*S/D
 3430   !--- Bildschirmaufbau ---
 3440   CLEAR #4
 3450   DISPLAY #4:AT 1 FROM 1 TO 2
5
 3460   PLOT #4:200,450+R2;200,450-
R2;1200,450-R2;1200,450+R2;200,450+
R2
 3470   !--- Beginn Simulation ---
 3480   FOR I=0 TO PI/2 STEP PI/180

 3490     PLOT #4:I*S+200,SIN(I)*R2
+450
 3500   NEXT I
 3510   FOR I=PI/2 TO PI/2*3 STEP P
I/30
 3520     PLOT #4:I*S+200,SIN(I)*R2
+450
 3530   NEXT I
 3540   FOR I=PI/2*3 TO PI*2 STEP P
I/180
 3550     PLOT #4:I*S+200,SIN(I)*R2
+450
 3560   NEXT I
 3570   CALL WAIT
 3580 END DEF
 3590 !
 3600 !**** Ausgabe SB I ****
 3610 !
 3620 DEF AUSGABE3
 3630   CLEAR TEXT
 3640   DISPLAY #5:AT 25 FROM 1 TO
1
 3650   DISPLAY #102:AT 1 FROM 1 TO
24
 3660   CALL DRUCK
 3670   CLEAR TEXT
 3680   PRINT AT 3,18:"Daten"
 3690   PRINT AT 4,18:"====="
 3700   PRINT AT 6,13:"vx: ";VX
 3710   PRINT AT 7,14:"r: ";R
 3720   PRINT AT 8,14:"d: ";D
 3730   PRINT AT 9,14:"B: ";B
 3740   PRINT AT 10,0:"Einfallswink
el: ";WA/PI*180
 3750   IF DF=1 THEN COPY
 3760   CALL WAIT
 3770 END DEF
 3780 !
 3790 !**** Eingabe SB II ****
 3800 !
 3810 DEF EINGABE4
 3820   CALL DESK1
 3830   CALL V_
 3840   CALL WA_
 3850   CALL BW_
 3860   CALL B_
 3870   CALL Q_
 3880   CALL M_
 3890   CALL FA_
 3900 END DEF
 3910 !
 3920 !**** Schraubenbahn II ****
 3930 !
 3940 DEF SBII
 3950   !--- Initialisierung ---
 3960   LET R=(M*SIN(WA)*V)/(Q*B)
 3970   IF R=0 THEN LET R2=0:LET S=
16:GOTO 4040
 3980   LET VX=COS(WA)*V
 3990   LET D=(VX*2*PI*M)/(B*Q)
 4000   LET H=INT(LOG10(R))
 4010   LET R2=R*10^(-H)*40:LET S=D
/2/PI*10^(-H)*40
 4020   !--- Bildschirmaufbau ---
 4030   CLEAR #4
 4040   DISPLAY #4:AT 1 FROM 1 TO 2
5
 4050   !--- Beginn der Simulation
----
 4060   FOR I=0 TO 6*PI STEP PI/180

 4070     LET X=COS(I)*R2*COS(BW)+4
00+S*I*SIN(BW)*FA:LET Y=SIN(I)*R2+4
50
 4080     IF X<0 OR X>1300 OR Y<0 O
R Y>899 THEN 4130
 4090     PLOT #4:X,Y
 4100     LET A$=INKEY$
 4110     IF A$="s" THEN EXIT FOR
 4120   NEXT I
 4130 END DEF
 4140 !
```

```
 4150 !
 4160 !**** B-Kreuze setzen ****
 4170 !
 4180 DEF BPLOT
 4190   FOR I=G1 TO G2 STEP 100
 4200     FOR I2=G3 TO G4 STEP 100
 4210       PLOT #4:I,I2,
 4220       PRINT #4:"x";
 4230     NEXT I2
 4240   NEXT I
 4250 END DEF
 4260 !
 4270 !**** Ausdruck ****
 4280 !
 4290 DEF DRUCK
 4300   LET DF=0
 4310   PRINT AT 10,5:"Ausdruck erw
ünscht (j/n)?"
 4320   CALL WAIT
 4330   IF A$="j" THEN LET DF=1:GOT
O 4350
 4340   IF A$<>"n" THEN 4320
 4350 END DEF
 4360 !
 4370 !**** Eingabedisplay ****
 4380 !
 4390 DEF DESK1
 4400   CLEAR TEXT
 4410   DISPLAY #102:AT 3 FROM 3 TO
 24
 4420   PRINT AT 5,14:"Eingabe"
 4430   PRINT AT 6,14:"======="
 4440   PRINT :PRINT :PRINT
 4450 END DEF
 4460 !
 4470 !**** Tastendruck ****
 4480 !
 4490 DEF WAIT
 4500   LET A$=INKEY$
 4510   IF A$="" THEN 4500
 4520 END DEF
 4530 !
 4540 !**** Eingaben ****
 4550 !
 4560 DEF D_
 4570   PRINT :INPUT PROMPT "Platte
nabstand:       ":D
 4580   LET FA=800/D
 4590 END DEF
 4600 DEF E_
 4610   PRINT :INPUT PROMPT "Konden
satorspannung:  ":UK
 4620   LET E=UK/D
 4630 END DEF
 4640 DEF B_
 4650   PRINT :INPUT PROMPT "Stärke
 des B-Feldes:  ":B
 4660 END DEF
 4670 DEF Q_
 4680   PRINT :INPUT PROMPT "Ladung
 des Teilchens: ":Q
 4690 END DEF
 4700 DEF M_
 4710   PRINT :INPUT PROMPT "Masse
des Teilchens:  ":M
 4720 END DEF
 4730 DEF VX_
 4740   PRINT :INPUT PROMPT "v in x
 - Richtung:    ":VX
 4750 END DEF
 4760 DEF FA_
 4770   PRINT :PRINT TAB(22);FA:PRI
NT CHR$(176);CHR$(176)
 4780   INPUT PROMPT "Darstellungsf
aktor:   ":FA
 4790 END DEF
 4800 DEF V_
 4810   PRINT :INPUT PROMPT "Geschw
indigkeit:      ":V
 4820 END DEF
 4830 DEF WA_
 4840   PRINT :INPUT PROMPT "Winkel
 zum B-Feld:    ":WA
 4850   LET WA=WA*PI/180
 4860 END DEF
 4870 DEF BW_
 4880   PRINT :INPUT PROMPT "Betrac
htungswinkel:   ":BW
 4890   LET BW=BW*PI/180
 4900 END DEF
```

Räuber-Beute-Population

Programmlänge: 3507 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128

Simulationen biologischer Zusammenhänge gewinnen in letzter Zeit immer mehr an Bedeutung. Die fortschreitende Beeinflußung der Umwelt durch den Menschen bleibt nicht ohne Folgen. Nur sind derartige Zusammenhänge nicht immer offensichtlich und nur über Jahre hinweg

zu erkennen. Ein Paradebeispiel für das Zusammenwirken vieler Faktoren, die in ihrer Summe den Begriff 'Natur' verkörpern, ist die Lebensgemeinschaft von Raubtieren und ihrer Beute.
Vor mehreren Jahren machten Biologen in Kanada ein Experiment, daß diese Zusammenhänge aufdeckte. Auf einer kleinen Insel, die sich in einem See befand, lebte eine Elch-Population. Da sie dort keinerlei natürliche Feinde besaß, vermehrte sie sich ungehindert bis allmählich die Vegetation derart in Mitleidenschaft gezogen wurde, daß sie die Elche nicht mehr versorgen konnte. Dieses typische Beispiel von Überweidung stellte also die weitere Existenz der Elche in Frage. Die einzige Rettung wäre der gezielte Abschuß gewesen. Doch kosten solche Maßnahmen nicht nur Geld sondern auch Zeit, denn schließlich müßte die Herdengröße ständig auf diese 'unnatürliche' Weise konstant gehalten werden. Die Lösung war schließlich der Aussatz von Wölfen auf dem Eiland mit einem überraschenden Ergebnis: Die Elchpopulation schrumpfte auf ein gesundes Maß, die Vegetation erholte sich und auch das Wolfsrudel behielt seine konstante Größe.
Ähnliche Ergebnisse zeigten auch die über neun Jahrzehnte geführten Aufzeichnungen der Hudson-Bay-Company von gejagten Schneehasen- und Luchsfellen.
Diese Zusammenhänge des biologischen Gleichgewichts lassen sich nun hervorragend auch auf einem Computer darstellen, was außerdem die Auswirkungen von Parameter-Variationen deutlich vor Augen führt. Den Berechnungen liegt das mathematische Modell der Dynamik von Räuber-Beute-Systemen nach Vito Volterrra und Alfred Lottka. Natürlich konnten in der vorliegenden Simulation nicht alle natürlichen Gegebenheiten berücksichtigt werden, die ein reales Zusammenspiel der Umweltfaktoren siumliert. So wurden der Übersichtlichkeit wegen nur die Zuwachs- und Abnahmeraten der Tiere berücksichtigt. Klimatische Faktoren und Vegetationsveränderungen lassen aber noch genügend Raum für eigene Programmerweiterungen.

Programmerläuterung

In diesem Simulations-Programm wurden als Räuber Füchse und als Beute Hasen eingesetzt. Nach dem Programmstart wird die Anzahl der Räuber und der Beute abgefragt (sog. Zustandsgrößen), deren Maximalwert in Abhängigkeit von der Bildschirmauflösung gewählt wurde.
Die Ausgabe der errechneten Daten kann nun in Tabellenform oder als Grafik (Phasenbild) erfolgen. In beiden Fällen wird der Berechnungsvorgang durch Eingabe von 'Q' abgebrochen. Über die Taste 'P' läßt sich die Grafik auch ausdrucken.
Die Zeiträume über die sich die Simulation hinzieht, werden im Programm in Jahren gerechnet. Dieser Wert bezieht sich auf die verwendeten Tierarten und errechnet sich aus den jeweiligen Vermehrungsraten. Für die Hasenpopulation ergibt sich eine Nettozuwachsrate bei ungehinderten Wachstum von 0.08 pro Woche. Bei den Füchsen beträgt der Erhaltungsbedarf 0.2 pro Woche. Konkret bedeutet dies, daß ein Fuchs 20% seines Eigengewichts pro Woche als Nahrung benötigt.
Bei anderen Tierarten (beispielsweise Großwild) müßten dann entsprechend der geringeren Vermehrungsgeschwindigkeit andere Konstanten für die Nettozuwachsrate der Beute und der Respirationsrate der Jäger eingesetzt werden.
Erwähnenswert sind auch folgende Besonderheiten des Programms:
Ab Zeile 510 wird ein Exception-(Ausnahme-) Konstrukt eingesetzt, um den Programmabbruch während der Graphikausgabe durch Verlassen des Bildschirms zu verhindern. Der zugehörige Handler namens 'Rechenüberlauf' befindet sich ab Zeile 1550.
Die Datenausgabe in der Graphik erfolgt über eine USING-Anweisung mit dazugehöriger IMAGE-Zeile. Alte Werte werden mit LOESCH$ gelöscht.

Reiner Kunz

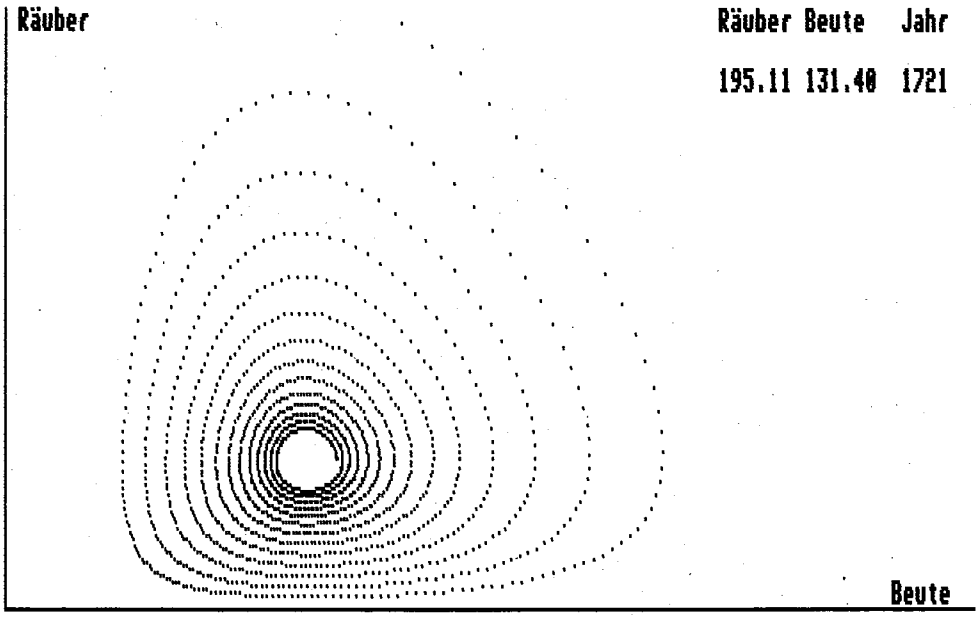
```
  100 PROGRAM "Räuber-Beute-Beziehu
ng"
  110 !
  120 TEXT 40
  130 PRINT AT 3,10:"Räuber-Beute-B
eziehung"
  140 PRINT AT 4,10:"==============
========"
  150 !
  160 PRINT AT 9,2:"Diese Simulatio
n zeigt grafisch den"
  170 PRINT AT 11,2:"Zusammenhang v
on der Anzahl der"
  180 PRINT AT 13,2:"Räuber und der
 Beutetiere in bezug"
  190 PRINT AT 15,2:"auf die Entwic
klung einer Population"
  200 !
  210 ! ******************
  220 ! * Programmstart *
  230 ! ******************
  240 !
  250 LET ZURAEU=.001 ! Zuwachsrate
 Räuber
  260 LET ZUBEUT=.025 ! Zuwachsrate
 Beute
  270 LET ABRAEU=.1 ! Abnahmerate R
äuber
  280 LET ABBEUT=.0005 ! Abnahmerat
e Beute
  290 LET JAHR=0
  300 LET LOESCH$=CHR$(142) ! Loesc
hstring für Graphikdatenzeile
  310 DO
  320   LET LOESCH$=LOESCH$&CHR$(14
2)
  330 LOOP UNTIL LEN(LOESCH$)=20
  340 !
  350 INPUT AT 18,2,PROMPT "Räuber
(10-199) ":X
  360 IF (X<10) OR(X>199) THEN GOTO
 350
  370 INPUT AT 20,2,PROMPT "Beute
(10-319) ":Y
  380 IF (Y<10) OR(Y>319) THEN GOTO
 370
  390 PRINT AT 22,2:"Tabelle/Grafik
 (T/G) ? ";
  400 DO
  410   LET ANT$=UCASE$(INKEY$)
  420 LOOP UNTIL ANT$="T" OR ANT$="
G"
  430 PRINT ANT$
  440 IF ANT$="T" THEN GOTO 900
  450 TEXT 80
  460 !
  470 ! ********************
  480 ! * Graphik-Ausgabe *
  490 ! ********************
  500 !
  510 WHEN EXCEPTION USE RECHENUEBE
RLAUF
  520   GRAPHICS HIRES 2
  530   SET PALETTE BLACK,GREEN
  540   PRINT #101,AT 1,2:"Räuber"
  550   PRINT #101,AT 20,74:"Beute"

  560   PLOT #101:0,0;0,719
  570   PLOT #101:0,0;1279,0
  580   !
  590   PLOT #101:(Y*4),(X*3.6)
  600   !
  610   PRINT #101,AT 1,60:"Räuber
Beute   Jahr"
  620   SET #101:INK 0
  630   PRINT #101,AT 3,59:LOESCH$
  640   SET #101:INK 1
  650   PRINT #101,AT 3,59,USING 10
60:X;
  660   PRINT #101,USING 1060:Y;
  670   PRINT #101,USING 1070:JAHR
  680   !
  690   GOSUB 790 ! Neue Berechnung

  700   PLOT #101:(Y*4),(X*3.6)
  710   LET ANT$=UCASE$(INKEY$)
  720   IF ANT$="Q" THEN GOTO 1360
  730   IF ANT$="P" THEN EXT "VDUMP
"
  740   IF (X=0) OR(Y=0) THEN GOTO
1130
  750   GOTO 620
  760 END WHEN
  770 END
  780 !
  790 ! ********************
  800 ! * Neue Berechnung *
  810 ! ********************
  820 !
  830 LET X1=(ZURAEU*Y*X-ABRAEU*X)+
X
  840 LET Y=(ZUBEUT*Y-ABBEUT*X*Y)+Y

  850 LET X=INT(X1*100)/100
  860 LET Y=INT(Y*100)/100
  870 LET JAHR=JAHR+1
  880 RETURN
  890 !
  900 ! ************************
  910 ! * Tabelle ausgeben *
  920 ! ************************
  930 !
  940 CLEAR SCREEN
  950 SET VIDEO MODE 0
```

```
  960 SET VIDEO COLOUR 0
  970 SET VIDEO X 40
  980 SET VIDEO Y 3
  990 OPEN #1:"VIDEO:"
 1000 SET #1:CURSOR CHARACTER 32
 1010 DISPLAY #1:AT 1 FROM 1 TO 3
 1020 PRINT #1:"TABELLE:";
 1030 PRINT #1:TAB(13);"RÄUBER  BEU
TE  JAHR"
 1040 !
 1050 PRINT AT 4,0:
 1060 IMAGE:####.##
 1070 IMAGE:######
 1080 PRINT TAB(12);
 1090 PRINT USING 1060:X;
 1100 PRINT USING 1060:Y;
 1110 PRINT USING 1070:JAHR
 1120 GOSUB 790
 1130 IF X=0 THEN
 1140   PRINT
 1150   PRINT "Räuber sind ausgesto
rben. Berechnung  kann nicht fortg
eführt werden."
 1160   GOTO 1330
 1170 END IF
 1180 IF Y=0 THEN
 1190   PRINT
 1200   PRINT "Beute ist ausgestorb
en. Berechnung kann nicht fortgefüh
rt werden."
 1210   GOTO 1330
 1220 END IF
 1230 !
 1240 LET ANT$=UCASE$(INKEY$)
 1250 !
 1260 IF NOT(ANT$="Q") THEN
 1270   GOTO 1080
 1280 ELSE
 1290   PRINT
 1300   GOTO 1390
 1310 END IF
 1320 !
 1330 ! ********************
 1340 ! * Neue Simulation *
 1350 ! ********************
 1360 !
 1370 PRINT
 1380 PRINT
 1390 PRINT "Wollen Sie eine neue S
imulation ? ";
 1400 DO
 1410   LET ANT$=UCASE$(INKEY$)
 1420 LOOP UNTIL ANT$="N" OR ANT$="
J"
 1430 PRINT ANT$
 1440 !
 1450 IF ANT$="J" THEN RUN
 1460 !
 1470 CLOSE #255
 1480 TEXT 80
 1490 END
 1500 !
 1510 ! ******************
 1520 ! * Fehlerabfang *
 1530 ! ******************
 1540 !
 1550 HANDLER RECHENUEBERLAUF
 1560   PING
 1570   IF EXTYPE=9216 THEN
 1580     PRINT "Rechenüberlauf !"
 1590   ELSE
 1600     PRINT "nicht lokalisierba
rer Fehler !"
 1610   END IF
 1620   GOTO 1380
 1630 END HANDLER
```

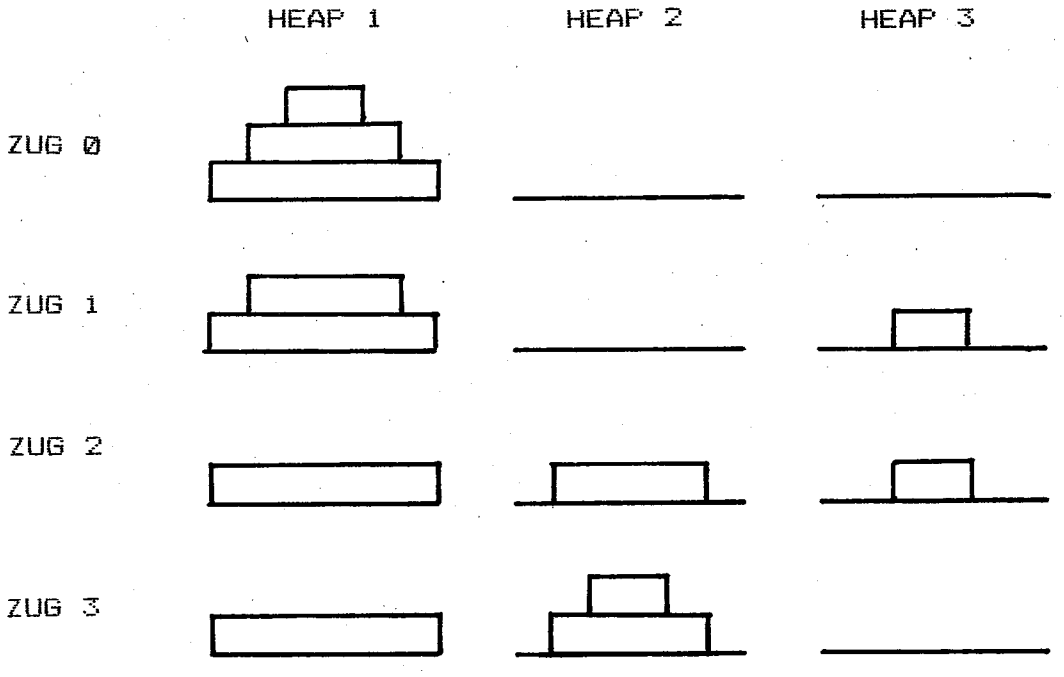


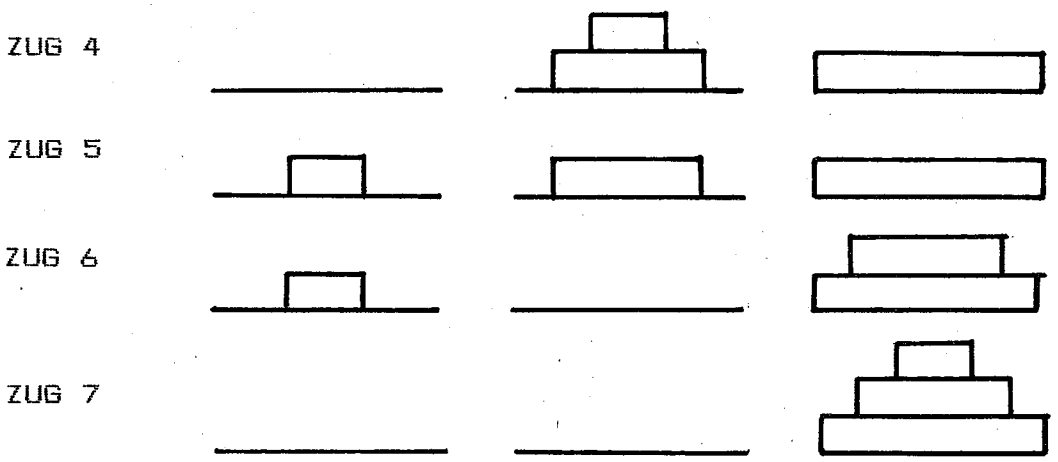
Hanoi-Türme in IS-BASIC

Programmlänge: 675 Bytes
Sprache: IS-BASIC
Zubehör: Enterprise 128

Rekursion, Rekursion. Die Domäne der komplexeren Hochsprachen wie Pascal und Algol wird nun mit IS-BASIC erstmals für Heimcomputer zugänglich. Wo andere Rechner nur mit Peek und Poke weiterkommen, braucht sich der Enterprise-Programmierer keine weiteren Gedanken zu machen. Das komfortable BASIC-Modul nimmt ihm die ganze Denkarbeit ab.
Was ist nun aber Rekursion, die von den einen geliebt und von den anderen verdammt wird ?
Niklaus Wirth, der Vater von Pascal und Modula-2, schreibt hierzu: 'Ein Objekt heißt rekursiv, wenn es sich selbst als Teil enthält oder mit Hilfe von sich selbst definiert ist.'
Als Standardbeispiel hierfür wird meist vom Bild im Bild gesprochen, oder dem optischen Verwirrspiel, das sich ergibt, wenn man zwei Spiegel so hält, daß man in den entstehenden Tunnel blicken kann.
Um nun Rekursion ermöglichen zu können benötigt man ein Prozedur-Konstrukt, welches in der Lage ist, lokale Variablen von globalen zu unterscheiden.
Als Beispiel für diese den meisten BASIC-Programmierern unbekannte Programmiermethodik wurde das folgende Programm 'Türme von Hanoi' geschrieben, welches die jeweils optimalen Zugfolgen beim Versetzen der Scheiben rekursiv berechnet.
Als kleine Spielerei werden neben den Zugfolgen auch der benötigte Speicherplatz angezeigt. So kann man sehr schön beobachten, wie der Adreßraum je nach Inkarnationslage pulsiert und die Ablageadressen der entsprechenden lokalen Variablen dynamisch erzeugt und wieder gelöscht werden.
Als Anreiz für eigene Versuche auf diesem Gebiet könnte man zum Beispiel Unterprogramme schreiben, die den momentanen Zustand des Turmversatzes grafisch auf den Bildschirm ausgeben.
Ebenfalls als Beispiele für die Verwendung von Rekursion in der Garfik seien die Programme 'Hilbert' und 'Sierpinski' genannt, die abhängig vom Schleifenindex unterschiedlich komplexe Graphiken erstellen.

Reiner Kunz






```
START
Wieviele Scheiben: 3

Scheibe von      nach    heap(1) heap(2) heap(3) Memory
 1        1  ->  3        2       0       1       43319
 2        1  ->  2        1       1       1       43394
 1        3  ->  2        1       2       0       43319
 3        1  ->  3        0       2       1       43469
 1        2  ->  1        1       1       1       43319
 2        2  ->  3        1       0       2       43394
 1        1  ->  3        0       0       3       43319
ok
copy
```

```
  100 PROGRAM "Hanoi"
  110 !
  120 DEF PUT(NUMBER,FROM,TO,BY)
  130   IF NUMBER>0 THEN
  140     CALL PUT(NUMBER-1,FROM,BY
,TO)
  150     LET HEAP(FROM)=HEAP(FROM)
-1
  160     LET HEAP(TO)=HEAP(TO)+1
  170     PRINT NUMBER,FROM;" -> ";
TO,HEAP(1),HEAP(2),HEAP(3),FREE
  180     CALL PUT(NUMBER-1,BY,TO,F
ROM)
  190   END IF
  200 END DEF
  210 !
  220 NUMERIC HEAP(1 TO 3)
  230 TEXT 80
  240 !
  250 INPUT PROMPT "Wieviele Scheib
en: ":N
  260 LET HEAP(1)=N !Initialisierun
g der Türme 1 bis 3
  270 LET HEAP(2)=0
  280 LET HEAP(3)=0
  290 PRINT
  300 PRINT "Scheibe","von","nach",
"heap(1)","heap(2)","heap(3)","Memo
ry"
  310 CALL PUT(N,1,3,2) !rekursiver
 Aufruf der Versetz-Prozedur
  320 !
  330 END
```

Logic

Rechner: Enterprise 64/128
Programmname: Logic
Programmlänge: ca. 9500 Bytes
Programmiersprache: BASIC
Zubehör: ---

Falls Sie mit dem Spiel "Superhirn" vertraut sind, wird Ihnen dieses Spiel keine Schwierigkeiten bereiten; aber auch Neulinge werden sich schnell für dieses Kombinations-

spiel begeistern: Hier gilt es, eine vom Computer erstellte Farbkombination zu ermitteln (dabei ist keine Farbe öfter als einmal vorhanden).

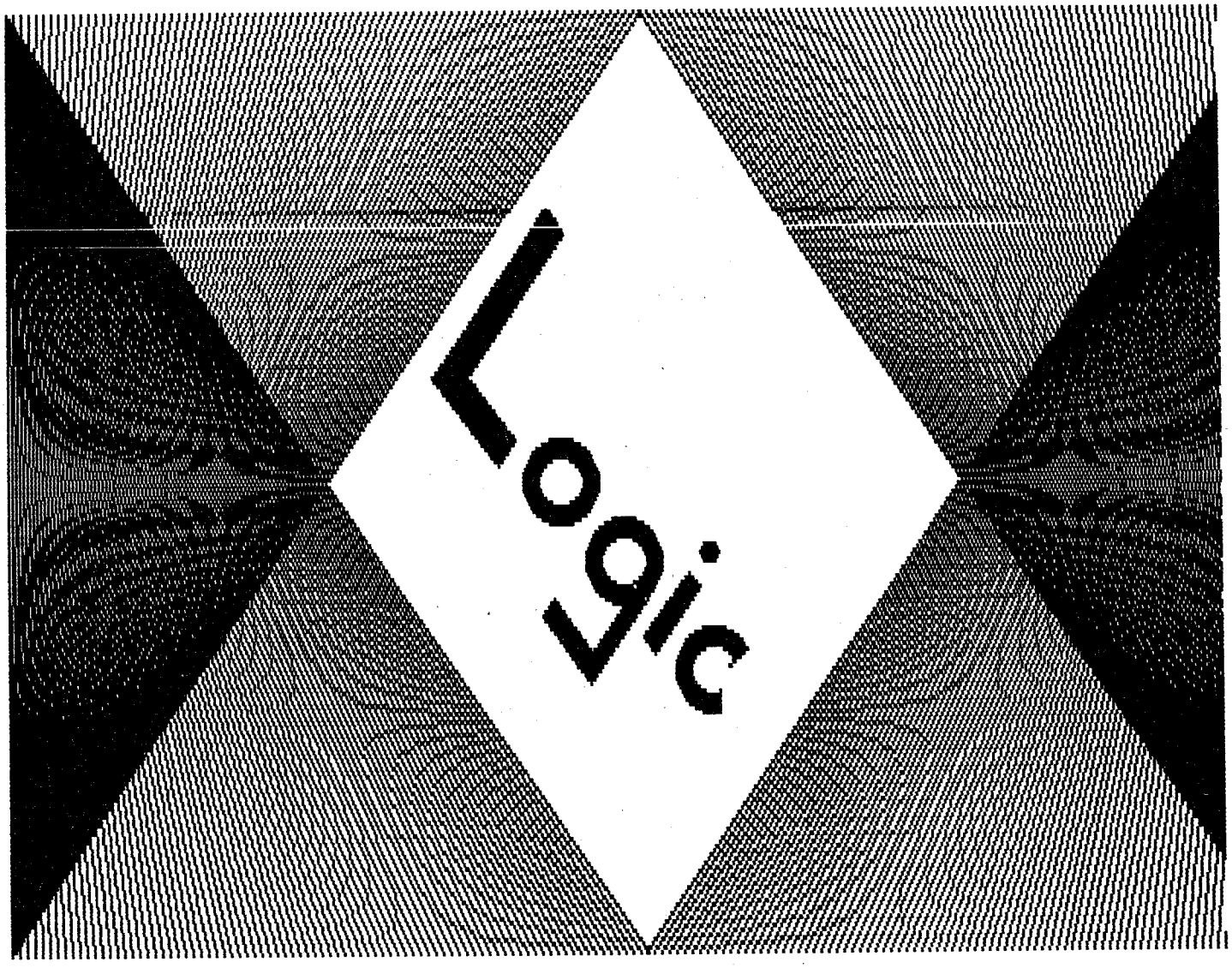
Bedienung des Programmes

Nach dem Start benötigt das Programm einige Zeit für die Initialisierung der Variablen sowie den Aufbau des Titelbildes. Nachdem dies geschehen ist, ertönt ein Klingelzeichen und das Titelbild erscheint solange in wechselnden Farbtönen, bis eine beliebige Taste gedrückt wird. Nun erscheint das Menü auf dem Bildschirm; Sie können jetzt zwischen folgenden drei Schwierigkeitsstufen wählen:

	Spalten	Versuche	Farben
Easy	4	8	6
Medium	5	10	8
Hard	6	12	10

Hierauf sind Sie im Spielmodus. Dabei bieten sich Ihnen folgende Alternativen:

* Cursor-Joystick links bzw. rechts
  Ihr "Cursor" (Kreuz) bewegt sich in der aktuellen Richtung;bereits gesetzte Farben werden hiervon nicht beeinflußt.
  Beachten Sie dabei bitte,daß zwei Felder rechts in Ihrem Spielfeld für die Anzeige der Resultate benötigt werden; diese können deshalb nicht verwendet werden.
* Cursor-Joystick oben bzw. unten



Der Zeiger neben der Farbskala (rechts am Bildschirm) bewegt sich.

* <ENTER>
Hiermit schließen Sie Ihre Eingabe ab. Sollten jedoch nicht alle Positionen belegt sein, so verfährt der Rechner, als ob Sie "1" gedrückt hätten. Ist die Eingabe aber korrekt, so zeigt der Computer die Anzahl der richtigen Farben, die jedoch in der falschen Position sind (weiße Zahl), sowie die Anzahl der in Farbe und Position mit dem Code übereinstimmenden Stecker (rote Zahl) an.

* "1"
Dieser Befehl löscht alle Ihre Eingaben dieses Versuchs und setzt den Cursor sowie den Skalenpfeil zurück.

* "s"
Dies ist Ihr "Aufgeben":Nun zeigt Ihnen der Rechner die Kombination. Nachdem Sie eine beliebige Taste gedrückt haben, erscheint das "Looser-Display". Nach erneutem Tastendruck befinden Sie sich wieder im Menü.

Haben Sie den Code auch mit Ihrem letzten Versuch nicht ermittelt, so zeigt Ihnen das Programm die Lösung und wartet auf einen Tastendruck (so können Sie Ihre Gedankengänge überprüfen). Nach dem "Winner-Display" befinden Sie sich wieder im Menü.

Viel Spaß beim Kombinieren!

Thomas Müller

```
100 !Logic
110 !für Chip Special
120 !von Thomas Müller
130 !
140 GOTO 280
150 CALL MENUE
160 IF AS<4 THEN
170   LET S=AS
180   CALL SPIEL
190   GOTO 150
200 END IF
210 !--- Ende ---
220 CLOSE #2
230 CLOSE #3
240 CLOSE #4
250 TEXT 40
260 END
270 !
280 !**** Initialisierung ****
290 !
300 CLEAR TEXT:SET STATUS OFF
305 PRINT AT 12,15:"Bitte warten"
310 DIM SP(6),L(3),M(3),F(3),C(6)
,E(6)
320 LET AS,S,X1,X2,Y1,Y2,FA,PO,FF
=0:LET A$=""
330 LET L(1)=4:LET L(2)=5:LET L(3
)=6
340 LET M(1)=8:LET M(2)=10:LET M(
3)=12
350 LET F(1)=6:LET F(2)=8:LET F(3
)=10
360 !
370 !--- Überschrift ---
380 SET VIDEO MODE 5
390 SET VIDEO COLOUR 2
400 SET VIDEO X 42
410 SET VIDEO Y 5
420 OPEN #2:"video:"
430 SET #2:PALETTE BLACK,RGB(.8,.
5,.8),0,0,0,0,0,0
440 !--- "Spieldisplay" ---
450 SET VIDEO MODE 1
460 SET VIDEO COLOUR 2
470 SET VIDEO X 42
480 SET VIDEO Y 25
490 OPEN #3:"video:"
500 !--- Leerseite ---
510 SET VIDEO MODE 0
520 SET VIDEO COLOUR 0
530 SET VIDEO X 42
540 SET VIDEO Y 25
550 OPEN #4:"video:"
560 SET #4:CURSOR CHARACTER 32
570 SET #102:PALETTE BLACK,BLUE,R
ED,0,0,0,0,0
580 SET #3:CHARACTER 64,0,24,60,1
26,126,60,24,0,0
590 SET #3:CHARACTER 150,240,248,
252,254,255,254,252,248,240
600 SET #3:CHARACTER 151,0,66,36,
24,24,36,66,0,0
610 SET #3:CHARACTER 152,0,255,25
5,255,255,255,255,0,0
620 SET #3:PALETTE BLACK,BLUE,RGB
(.4,.1,.4),YELLOW,RGB(.7,.5,.7),MAG
ENTA,CYAN,WHITE
```

```
  630 SET #3:BIAS RGB(.5,.4,0)
  640 !---- Titelbild ----
  650 SET VIDEO MODE 1
  660 SET VIDEO COLOUR 0
  670 SET VIDEO X 42
  680 SET VIDEO Y 25
  690 OPEN #1:"video:"
  700 CLEAR #1
  710 FOR I=0 TO 672 STEP 8
  720   PLOT #1:0,0;672-I,899,I+671
,899;1343,0
  730   PLOT #1:0,899;672-I,0,1343,
899;I+671,0
  740 NEXT I
  750 OPTION ANGLE DEGREES
  760 !-- L --
  770 PLOT #1:450,550;ANGLE 53;FORW
ARD 200;RIGHT 104;FORWARD 30;RIGHT
76;FORWARD 170;LEFT 76;FORWARD 80;
  780 PLOT #1:RIGHT 76;FORWARD 30;R
IGHT 104;FORWARD 110
  790 !-- o --
  800 PLOT #1:585,450,ELLIPSE 40,40
,ELLIPSE 25,25,
  810 !-- g --
  820 PLOT #1:650,375,ELLIPSE 40,40
,ELLIPSE 25,25,
  830 PLOT #1:675,338;ANGLE 233;FOR
WARD 100;RIGHT 104;FORWARD 80;RIGHT
 76;FORWARD 20;RIGHT 104;FORWARD 65
;
  840 PLOT #1:LEFT 104;FORWARD 80
  850 !-- i --
  860 PLOT #1:670,300;ANGLE 53;FORW
ARD 80;RIGHT 104;FORWARD 20;RIGHT 7
6;FORWARD 80;RIGHT 104;FORWARD 20
  870 PLOT #1:740,380,ELLIPSE 10,10
,
  880 !-- c --
  890 PLOT #1:745,270,ELLIPSE 40,40
,ELLIPSE 25,25,
  900 PLOT #1:770,290,
  910 SET #1:INK 0
  920 PRINT #1:CHR$(159);
  930 PLOT #1:760,275,
  940 PRINT #1:CHR$(159);
  950 PLOT #1:768,260,
  960 PRINT #1:CHR$(159);
  970 PLOT #1:752,260,
  980 PRINT #1:CHR$(159);
  990 SET #1:INK 1
 1000 PLOT #1:780,290;760,275,750,2
35;750,250
 1010 !-- Füllen --
 1020 PLOT #1:460,550,PAINT,585,480
,PAINT
 1030 PLOT #1:655,410,PAINT,660,320
,PAINT
 1040 PLOT #1:680,305,PAINT,740,385
,PAINT,740,240,PAINT
 1050 DISPLAY #1:AT 1 FROM 1 TO 25
 1060 PING
 1070 PING
 1080 LET A$=INKEY$
 1090 IF A$<>"" THEN 1120
 1100 SET #1:PALETTE RGB(RND,RND,RN
D),RGB(RND,RND,RND),0,0,0,0,0,0
 1110 GOTO 1080
 1120 GOTO 150
 1130 !
 1140 !**** Menü ****
 1150 !
 1160 DEF MENUE
 1170   DISPLAY #2:AT 1 FROM 1 TO 5

 1180   DISPLAY #3:AT 6 FROM 6 TO 2
5
 1190   CLEAR #2
 1200   CLEAR #3
 1210   PLOT #2:350,170,
 1220   PRINT #2:"Logic"
 1230   PLOT #2:350,80;950,80
 1240   PLOT #3:350,550,
 1250   SET #3:INK 1
 1260   PRINT #3:"1 <= Easy"
 1270   PLOT #3:350,450,
 1280   SET #3:INK 2
 1290   PRINT #3:"2 <= Medium"
 1300   PLOT #3:350,350,
 1310   SET #3:INK 3
 1320   PRINT #3:"3 <= Hard"
 1330   PLOT #3:350,250,
 1340   SET #3:INK 4
 1350   PRINT #3:"4 <= End"
 1360   CALL WAIT
 1370   IF ORD(A$)<49 OR ORD(A$)>53
THEN 1360
 1380   LET AS=VAL(A$)
 1390 END DEF
 1400 !
 1410 !**** Spielverlauf ****
 1420 !
 1430 DEF SPIEL
 1440   !--- Spieldisplay ---
 1450   CLEAR #3
 1460   DISPLAY #3:AT 1 FROM 1 TO 2
5
 1470   SET #3:INK 8
 1480   FOR I=0 TO L(S)+2
 1490     PLOT #3:(I+2)*72,0;(I+2)*
72,M(S)*72
 1500   NEXT I
 1510   FOR I=0 TO M(S)
 1520     PLOT #3:2*72,I*72;(L(S)+4
```

```
)*72,I*72
 1530    NEXT I
 1540    FOR I=1 TO F(S)
 1550      PLOT #3:17.5*72,I*72+8,
 1560      SET #3:INK I
 1570      PRINT #3:CHR$(64);
 1580    NEXT I
 1590    SET #3:INK 8
 1600    FOR I=1 TO M(S)
 1610      PLOT #3:0,I*72-8,
 1620      PRINT #3,USING "##":I
 1630    NEXT I
 1640    PLOT #3:16.5*72-16,0;16.5*7
2-16,F(S)*72+16;1332,F(S)*72+16;133
2,0;16.5*72-16,0
 1650    FOR I=1 TO 5
 1660      PLOT #3:(10+I)*72,200,
 1670      SET #3:INK RND(16)+1
 1680      PRINT #3:"Logic"(I:I);
 1690    NEXT I
 1700    CALL CODEBOX
 1710    PLOT #3:(L(S)+6)*72,M(S)*72
-10,PAINT
 1720    !-- Code erstellen --
 1730    CALL CODE
 1740    !-- Versuche --
 1750    LET V=1
 1760    CALL LINERST
 1770    CALL XSET
 1780    CALL YSET
 1790    !--- Eingabe ---
 1800    CALL WAIT
 1810    LET AS=ORD(A$)
 1820    IF AS=176 THEN LET Y2=Y1:LE
T Y1=Y1+SGN(F(S)-Y1):CALL YSET:GOTO
 1800
 1830    IF AS=180 THEN LET Y2=Y1:LE
T Y1=Y1-SGN(Y1-1):CALL YSET:GOTO 18
00
 1840    IF AS=184 THEN LET X2=X1:LE
T X1=X1-SGN(X1-1):CALL XSET:GOTO 18
00
 1850    IF AS=188 THEN LET X2=X1:LE
T X1=X1+SGN(L(S)-X1):CALL XSET:GOTO
 1800
 1860    IF AS=32 THEN LET E(X1)=Y1:
LET X2=X1:CALL XSET:GOTO 1800
 1870    IF AS=108 THEN
 1880      FOR I=1 TO L(S)
 1890        PLOT #3:(I+1)*72+8,V*72
-8,
 1900        SET #3:INK 0
 1910        PRINT #3:CHR$(152);
 1920      NEXT I
 1930      CALL CLEARL
 1940      GOTO 1760
 1950    END IF
 1960    IF AS=13 THEN
 1970      LET FF=0
 1980      FOR I=1 TO L(S)
 1990        IF E(I)=0 THEN LET FF=1

 2000      NEXT I
 2010      IF FF=1 THEN 1880
 2020      SET #3:INK 0
 2030      PLOT #3:(X1+1)*72+8,V*72-
8,
 2040      PRINT #3:CHR$(152);
 2050      SET #3:INK E(X1)
 2060      PLOT #3:(X1+1)*72+8,V*72-
8,
 2070      PRINT #3:CHR$(64);
 2080      CALL TEST
 2090      PLOT #3:(L(S)+2)*72+8,V*7
2-8,
 2100      SET #3:INK WHITE
 2110      PRINT #3,USING "#":FA;
 2120      PLOT #3:(L(S)+3)*72+8,V*7
2-8,
 2130      SET #3:INK RED
 2140      PRINT #3,USING "#":PO;
 2150      IF PO=L(S) THEN CALL UP:G
OTO 2230
 2160      LET V=V+1
 2170      IF V>M(S) THEN CALL DOWN:
GOTO 2230
 2180      CALL CLEARL
 2190      GOTO 1760
 2200    END IF
 2210    IF AS=115 THEN CALL SHOW:GO
TO 2230
 2220    GOTO 1800
 2230 END DEF
 2240 !
 2250 !**** Richtig ****
 2260 !
 2270 DEF UP
 2280    CALL HAND
 2290    PLOT #3:600,620;ANGLE 60;FO
RWARD 80;LEFT 10;FORWARD 20;LEFT 80
;FORWARD 150;RIGHT 60;FORWARD 80;RI
GHT 70;FORWARD 50;RIGHT 20;FORWARD
200;
 2300    PLOT #3:RIGHT 20;FORWARD 50
;RIGHT 70;FORWARD 80;RIGHT 30;FORWA
RD 100;LEFT 50;FORWARD 80,700,700,P
AINT
 2310    SET #3:INK BLACK
 2320    FOR I=0 TO 5
 2330      PLOT #3:670,800,ELLIPSE(I
+5)*13,(I+1)*10,
 2340    NEXT I
 2350    SET #3:INK 8
 2360    PLOT #3:100,350,
```

```
 2370    PRINT #3:"Top!";
 2380    PLOT #3:100,250,
 2390    PRINT #3:"Sie haben es beim
"
 2400    PLOT #3:200,160,
 2410    PRINT #3:V;"-ten Versuch"
 2420    PLOT #3:300,70,
 2430    PRINT #3:"geschafft!"
 2440    DISPLAY #3:AT 1 FROM 1 TO 2
5
 2450    CALL WAIT
 2460 END DEF
 2470 !
 2480 !**** Verloren ****
 2490 !
 2500 DEF DOWN
 2510    CALL HAND
 2520    PLOT #3:600,280;ANGLE-60;FO
RWARD 80;LEFT-10;FORWARD 20;LEFT-80
;FORWARD 150;RIGHT-60;FORWARD 80;RI
GHT-70;FORWARD 50;RIGHT-20;FORWARD
200;
 2530    PLOT #3:RIGHT-20;FORWARD 50
;RIGHT-70;FORWARD 80;RIGHT-30;FORWA
RD 100;LEFT-50;FORWARD 80,650,250,P
AINT
 2540    SET #3:INK BLACK
 2550    FOR I=0 TO 5
 2560      PLOT #3:670,100,ELLIPSE(I
+5)*13,(I+1)*10,
 2570    NEXT I
 2580    SET #3:INK 8
 2590    PLOT #3:100,899,
 2600    PRINT #3:"Tja, da haben Sie
"
 2610    PLOT #3:100,800,
 2620    PRINT #3:"falsch kombiniert
!"
 2630    DISPLAY #3:AT 1 FROM 1 TO 2
5
 2640    CALL WAIT
 2650 END DEF
 2660 !
 2670 !**** Handrumpf ****
 2680 !
 2690 DEF HAND
 2700    CALL WAIT
 2710    DISPLAY #4:AT 1 FROM 1 TO 2
5
 2720    CLEAR #3
 2730    SET #3:INK 8
 2740    OPTION ANGLE DEGREES
 2750    FOR I=1 TO 4
 2760      PLOT #3:500,I*80+240+I*8;
ANGLE 45;FORWARD 45;RIGHT 45;FORWAR
D 300;RIGHT 45;FORWARD 45;RIGHT 45;
FORWARD 20;
 2770      PLOT #3:RIGHT 45;FORWARD
45;RIGHT 45;FORWARD 300;RIGHT 45;FO
RWARD 45;RIGHT 45;FORWARD 20,600,I*
80+240+I*8,PAINT
 2780    NEXT I
 2790 END DEF
 2800 !
 2810 !**** Code erstellen ****
 2820 !
 2830 DEF CODE
 2840    RANDOMIZE
 2850    FOR I=1 TO L(S)
 2860      LET C(I)=RND(F(S))+1
 2870      LET FF=0
 2880      FOR I2=1 TO I-1
 2890        IF C(I)=C(I2) THEN LET
FF=1
 2900      NEXT I2
 2910      IF FF=1 THEN 2860
 2920    NEXT I
 2930 END DEF
 2940 !
 2950 !**** Versuch überprüfen ****

 2960 !
 2970 DEF TEST
 2980    LET FA,PO=0
 2990    FOR I=1 TO L(S)
 3000      FOR I2=1 TO L(S)
 3010        IF E(I)=C(I2) THEN LET
FA=FA+1
 3020      NEXT I2
 3030      IF E(I)=C(I) THEN LET PO=
PO+1
 3040    NEXT I
 3050    LET FA=FA-PO
 3060 END DEF
 3070 !
 3080 !**** Lösung zeigen ****
 3090 !
 3100 DEF SHOW
 3110    SET #3:INK 0
 3120    PLOT #3:(L(S)+6)*72,M(S)*72
-10,PAINT
 3130    CALL CODEBOX
 3140    FOR I=1 TO L(S)
 3150      SET #3:INK 8
 3160      PLOT #3:(L(S)+4.5+I)*72,M
(S)*72;(L(S)+4.5+I)*72,(M(S)-1)*72
 3170      PLOT #3:(L(S)+3.5+I)*72+8
,M(S)*72-8,
 3180      SET #3:INK C(I)
 3190      PRINT #3:CHR$(64);
 3200    NEXT I
 3210    CALL WAIT
 3220 END DEF
 3230 !
```

```
 3240 !**** Tastendruck ****
 3250 !
 3260 DEF WAIT
 3270   LET A$=INKEY$
 3280   IF A$="" THEN 3270
 3290 END DEF
 3300 !
 3310 !**** Variablen einer Zeile r
ücksetzen ****
 3320 !
 3330 DEF LINERST
 3340   LET X1,X2,Y1,Y2=1
 3350   FOR I=1 TO 6
 3360     LET E(I)=0
 3370   NEXT I
 3380 END DEF
 3390 !
 3400 !**** Position verändern ****

 3410 !
 3420 DEF XSET
 3430   PLOT #3:(X2+1)*72+8,V*72-8,

 3440   SET #3:INK 0
 3450   PRINT #3:CHR$(152);
 3460   IF E(X2)<>0 THEN
 3470     SET #3:INK E(X2)
 3480     PLOT #3:(X2+1)*72+8,V*72-
8,
 3490     PRINT #3:CHR$(64);
 3500   END IF
 3510   PLOT #3:(X1+1)*72+8,V*72-8,
 3520   SET #3:INK 8
 3530   PRINT #3:CHR$(151);
 3540 END DEF
 3550 !
 3560 !**** Farbe ändern ****
 3570 !
 3580 DEF YSET
 3590   PLOT #3:16.5*72,Y2*72+8,
 3600   SET #3:INK 0
 3610   PRINT #3:CHR$(159);
 3620   PLOT #3:16.5*72,Y1*72+8,
 3630   SET #3:INK 8
 3640   PRINT #3:CHR$(150);
 3650 END DEF
 3660 !
 3670 !**** Initialisieren der Zeil
e ****
 3680 !
 3690 DEF CLEARL
 3700   FOR I=1 TO 6
 3710     LET E(I)=0
 3720   NEXT I
 3730   LET Y2=Y1:LET Y1=1:CALL YSE
T
 3740 END DEF
 3750 !
 3760 !**** Rahmen des Codes ****
 3770 !
 3780 DEF CODEBOX
 3790   SET #3:INK 8
 3800   PLOT #3:(L(S)+4.5)*72,M(S)*
72;(L(S)*2+4.5)*72,M(S)*72;(L(S)*2+
4.5)*72,(M(S)-1)*72;
 3810   PLOT #3:(L(S)+4.5)*72,(M(S)
-1)*72;(L(S)+4.5)*72,M(S)*72
 3820 END DEF
```

Skatrunde

Programmlänge: 12800 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128

Die Punktezählung beim Skat ist eine der weniger schönen Pflichten des beliebten Kartenspiels. Den Punktestand durch den Enterprise ausführen zu lassen, ist die bequemste Möglichkeit der Arbeitserleichterung.
'Skatschreiber' kann für drei Spieler die Punkteführung übernehmen. Dabei sind bis zu 50 Spiele möglich. Der Geber unter den drei Spielern wird durch zwei Pfeile vor und nach dem Namen angezeigt. Folgende Spiele sind möglich:

N :Null	NO:Null ouvert
R :Revolution	NH:Null-Hand
OH:Null ouvert-Hand	K :Kreuz
P :Pik	H :Herz
C :Karo	G :Grand

Bei der Eingabe spielen Groß- oder Kleinbuchstaben keine Rolle. Obwohl die Skatregeln bekannt sein dürften, sind doch einige Kleinigkeiten bei der Programmbedienung zu beachten. Nach Contra, Re, Bock und Zippe werden Bockrunden angezeigt. Unter der Spalte 'Spiel' wird die Art des Spieles mit Hilfe von Sym-

bolen dargestellt. So steht beispielsweise für Grand mit oder ohne 2 immer 'G2'.
Auf die Frage 'Wieviele Buben hat der Spieler' wird die Zahl der Buben eingegeben, egal, ob 'mit' oder 'ohne' gespielt wurde. Für ein Spiel 'mit 3' oder 'ohne 3' muß immer die '3' angegeben werden.
Gültigkeit hat folgende Punkterechnung: Gewinnt der Spieler, erhalten die Mitspieler die Spielpunkte auf ihr Konto angerechnet. Der Spieler hingegen behält seinen alten Punktestand. Sollte er jedoch verlieren, wird ihm der doppelte Spielwert zugeteilt, während seinen Kollegen die ursprünglichen Konten erhalten bleiben.
Somit ist Sieger, wer schließlich die wenigsten Punkte 'kassiert' hat.

Reiner Kunz

```
  100 PROGRAM "Skat-Runde"
  110 !
  120 TEXT 80
  130 PRINT "************ Spielberec
hnung ************"
  140 PRINT "          fuer 3 oder 4
 Spieler          "
  150 PRINT "************************
********************"
  160 PRINT
  170 INPUT PROMPT "Wieviele Spiele
r :":SPIELERZA
  180 IF SPIELERZA<3 OR SPIELERZA>4
 THEN GOTO 170
  190 !
  200 NUMERIC A(50),B(50),C(50),D(5
0)
  210 STRING SPANZEI$(50,20)*2
  220 !
  230 FOR ZX=0 TO 50
  240   LET A(ZX)=0
  250   LET B(ZX)=0
  260   LET C(ZX)=0
  270   LET D(ZX)=0
  280 NEXT ZX
  290 !
  300 FOR XY1=0 TO 50
  310   FOR XY2=0 TO 20
  320     LET SPANZEI$(XY1,XY2)=""
  330   NEXT XY2
  340 NEXT XY1
  350 !
  360 LET GEBER=1
  370 LET ZEILE=1
  380 LET P=0
  390 LET I$=""
  400 LET V1=0
  410 !
  420 PRINT :PRINT :PRINT
  430 PRINT "Geben Sie die Namen de
r Spieler ein!"
  440 INPUT PROMPT "1:":NAM1$
  450 INPUT PROMPT "2:":NAM2$
  460 INPUT PROMPT "3:":NAM3$
  470 !
  480 IF P=4 THEN
  490   INPUT PROMPT "4:":NAM4$
  500 END IF
  510 !
  520 ! ****************************
  530 ! * TABELLENKOPF ERSTELLEN *
  540 ! ****************************
  550 !
  560 IF SPIELERZA=4 THEN
  570   CLEAR SCREEN
  580   PRINT "NR";TAB(15);NAM1$;TA
B(22);NAM2$;TAB(29);NAM3$;TAB(36);N
AM4$;TAB(45);"SPIEL"
  590 END IF
  600 !
  610 IF SPIELERZA=3 THEN
  620   CLEAR SCREEN
  630   PRINT "NR";TAB(15);NAM1$;TA
B(25);NAM2$;TAB(35);NAM3$;TAB(45);"
SPIEL"
  640 END IF
  650 GOTO 3370
  660 !
  670 IF SPIELERZA=4 THEN GOTO 4250
  680 !
  690 ! ******************************
***
  700 ! * GEBERANZEIGE BEI 3 SPIELE
RN*
  710 ! ******************************
***
  720 !
  730 PRINT AT 1,14:" "
  740 PRINT AT 1,15+LEN(NAM1$):" "
  750 PRINT AT 1,24:" "
  760 PRINT AT 1,25+LEN(NAM2$):" "
  770 PRINT AT 1,34:" "
  780 PRINT AT 1,35+LEN(NAM3$):" "
  790 !
  800 IF GEBER=1 THEN
  810   PRINT AT 1,14:">"
  820   PRINT AT 1,15+LEN(NAM1$):"<
```

```
"
  830 END IF
  840 !
  850 IF GEBER=2 THEN
  860   PRINT AT 1,24:">"
  870   PRINT AT 1,25+LEN(NAM2$):"<
"
  880 END IF
  890 !
  900 IF GEBER=3 THEN
  910   PRINT AT 1,34:">"
  920   PRINT AT 1,35+LEN(NAM3$):"<
"
  930 END IF
  940 !
  950 LET GEBER=GEBER+1
  960 !
  970 IF GEBER=4 THEN LET GEBER=1
  980 !
  990 ! ************************
 1000 ! * WAS WIRD GESPIELT *
 1010 ! ************************
 1020 !
 1030 INPUT AT 20,1,PROMPT "Spieler
 :":SPIELER$
 1040 PRINT AT 20,30:SPIELER$
 1050 INPUT AT 21,1,PROMPT "Was wir
d gespielt :":SPART$
 1060 PRINT AT 21,30:SPART$
 1070 !
 1080 ! ********************
 1090 ! * PUNKTERECHNUNG *
 1100 ! ********************
 1110 ! * NULLSPIELE *
 1120 ! ****************
 1130 !
 1140 LET SPWERT=0
 1150 IF RTRIM$(UCASE$(SPART$))="N"
 THEN LET SPWERT=23
 1160 IF RTRIM$(UCASE$(SPART$))="NO
" THEN LET SPWERT=46
 1170 IF RTRIM$(UCASE$(SPART$))="R"
 THEN LET SPWERT=92
 1180 IF RTRIM$(UCASE$(SPART$))="NH
" THEN LET SPWERT=36
 1190 IF RTRIM$(UCASE$(SPART$))="OH
" THEN LET SPWERT=59
 1200 IF SPWERT>0 THEN GOTO 3660
 1210 !
 1220 ! ***************
 1230 ! * FARBSPIELE *
 1240 ! ***************
 1250 !
 1260 IF RTRIM$(UCASE$(SPART$))="K"
 THEN LET SPWERT=12
 1270 IF RTRIM$(UCASE$(SPART$))="P"
 THEN LET SPWERT=11
 1280 IF RTRIM$(UCASE$(SPART$))="H"
 THEN LET SPWERT=10
 1290 IF RTRIM$(UCASE$(SPART$))="C"
 THEN LET SPWERT=9
 1300 IF RTRIM$(UCASE$(SPART$))="G"
 THEN LET SPWERT=24
 1310 IF INKEY$="" THEN GOTO 1310
 1320 !
 1330 ! ******************
 1340 ! * SPIELABFRAGE *
 1350 ! ******************
 1360 !
 1370 GOSUB 3630
 1380 PRINT AT 21,1:"Wieviele Buben
 hat der Spieler :";
 1390 INPUT M$
 1400 GOSUB 3630
 1410 INPUT AT 21,1,PROMPT "Wurde H
and gespielt (j/n) :":D$
 1420 GOSUB 3630
 1430 INPUT AT 21,1,PROMPT "Ist ein
e Partei schneider (j/n) :":E$
 1440 GOSUB 3630
 1450 INPUT AT 21,1,PROMPT "Ist ein
e Partei schwarz (j/n) :":F$
 1460 GOSUB 3630
 1470 INPUT AT 21,1,PROMPT "Wurde c
ontra gegeben (j/n) :":G$
 1480 GOSUB 3630
 1490 INPUT AT 21,1,PROMPT "Wurde r
e gegeben (j/n) :":H$
 1500 GOSUB 3630
 1510 INPUT AT 21,1,PROMPT "Wurde B
ock gegeben (j/n) :":W$
 1520 GOSUB 3630
 1530 INPUT AT 21,1,PROMPT "Wurde Z
ippe gegeben (j/n) :":S$
 1540 GOSUB 3630
 1550 INPUT AT 21,1,PROMPT "Hat der
 Spielende gewonnen (j/n) :":K$
 1560 GOSUB 3630
 1570 INPUT AT 21,1,PROMPT "Haben S
ie einen Fehler gemacht (j/n) :":I$

 1580 IF NOT(RTRIM$(UCASE$(I$))="J"
) THEN GOTO 1620
 1590 LET GEBER=GEBER-1
 1600 GOTO 3370
 1610 !
 1620 ! ********************
 1630 ! * SPIELBERECHNUNG *
 1640 ! ********************
 1650 !
 1660 LET K=1
 1670 LET K=K+VAL(RTRIM$(M$))
 1680 IF RTRIM$(UCASE$(D$))="J" THE
N LET K=K+1
```

```
 1690 IF RTRIM$(UCASE$(E$))="J" THE
N LET K=K+1
 1700 IF RTRIM$(UCASE$(F$))="J" THE
N LET K=K+1
 1710 LET SPWERT=SPWERT*K
 1720 IF RTRIM$(UCASE$(G$))="J" THE
N LET SPWERT=SPWERT*2
 1730 IF RTRIM$(UCASE$(H$))="J" THE
N LET SPWERT=SPWERT*2
 1740 IF RTRIM$(UCASE$(W$))="J" THE
N LET SPWERT=SPWERT*2
 1750 IF RTRIM$(UCASE$(S$))="J" THE
N LET SPWERT=SPWERT*2
 1760 !
 1770 ! ***************
 1780 ! * ABRECHNUNG *
 1790 ! ***************
 1800 !
 1810 LET V=0
 1820 IF RTRIM$(UCASE$(G$))="J" THE
N LET V=V+3
 1830 IF RTRIM$(UCASE$(H$))="J" THE
N LET V=V+3
 1840 IF RTRIM$(UCASE$(W$))="J" THE
N LET V=V+3
 1850 IF RTRIM$(UCASE$(S$))="J" THE
N LET V=V+3
 1860 IF V<>0 THEN LET V1=V1+V
 1870 IF RTRIM$(UCASE$(K$))="" OR R
TRIM$(UCASE$(K$))="N" THEN GOTO 205
0
 1880 IF V1=0 THEN GOTO 1990
 1890 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM1$) THEN LET A(ZEILE)=A(ZEILE-1)+
2*SPWERT
 1900 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM2$) THEN LET B(ZEILE)=B(ZEILE-1)+
2*SPWERT
 1910 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM3$) THEN
 1920   LET C(ZEILE)=C(ZEILE-1)+2*S
PWERT
 1930   IF SPIELERZA=3 THEN GOTO 21
70
 1940 END IF
 1950 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM4$) THEN
 1960   LET D(ZEILE)=D(ZEILE-1)+2*S
PWERT
 1970   GOTO 2170
 1980 END IF
 1990 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM1$) THEN LET A(ZEILE)=A(ZEILE-1)+
SPWERT
 2000 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM2$) THEN LET B(ZEILE)=B(ZEILE-1)+
SPWERT
 2010 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM3$) THEN LET C(ZEILE)=C(ZEILE-1)+
SPWERT
 2020 IF SPIELERZA=3 THEN GOTO 2040

 2030 IF RTRIM$(SPIELER$)<>RTRIM$(N
AM4$) THEN LET D(ZEILE)=D(ZEILE-1)+
SPWERT
 2040 GOTO 2300
 2050 IF V1=0 THEN GOTO 2120
 2060 IF RTRIM$(SPIELER$)=RTRIM$(NA
M1$) THEN LET A(ZEILE)=A(ZEILE-1)+4
*SPWERT
 2070 IF RTRIM$(SPIELER$)=RTRIM$(NA
M2$) THEN LET B(ZEILE)=B(ZEILE-1)+4
*SPWERT
 2080 IF RTRIM$(SPIELER$)=RTRIM$(NA
M3$) THEN LET C(ZEILE)=C(ZEILE-1)+4
*SPWERT
 2090 IF SPIELERZA=3 THEN GOTO 2170

 2100 IF RTRIM$(SPIELER$)=RTRIM$(NA
M4$) THEN LET D(ZEILE)=D(ZEILE-1)+4
*SPWERT
 2110 GOTO 2170
 2120 IF RTRIM$(SPIELER$)=RTRIM$(NA
M1$) THEN LET A(ZEILE)=A(ZEILE-1)+2
*SPWERT
 2130 IF RTRIM$(SPIELER$)=RTRIM$(NA
M2$) THEN LET B(ZEILE)=B(ZEILE-1)+2
*SPWERT
 2140 IF RTRIM$(SPIELER$)=RTRIM$(NA
M3$) THEN LET C(ZEILE)=C(ZEILE-1)+2
*SPWERT
 2150 IF SPIELERZA=3 THEN GOTO 2300

 2160 IF RTRIM$(SPIELER$)=RTRIM$(NA
M4$) THEN LET D(ZEILE)=D(ZEILE-1)+2
*SPWERT
 2170 IF V1=0 THEN GOTO 2300
 2180 IF A(ZEILE)=0 THEN LET A(ZEIL
E)=A(ZEILE-1)
 2190 IF B(ZEILE)=0 THEN LET B(ZEIL
E)=B(ZEILE-1)
 2200 IF C(ZEILE)=0 THEN LET C(ZEIL
E)=C(ZEILE-1)
 2210 IF SPIELERZA=3 THEN GOTO 2230

 2220 IF D(ZEILE)=0 THEN LET D(ZEIL
E)=D(ZEILE-1)
 2230 LET V1=V1-1
 2240 PRINT AT 19,25:"BOCKRUNDEN:";
V1
 2250 IF SPIELERZA=3 THEN
 2260   GOTO 2330
 2270 ELSE
 2280   GOTO 1570
```

```
 2290 END IF
 2300 IF A(ZEILE)=0 THEN LET A(ZEIL
E)=A(ZEILE-1)
 2310 IF B(ZEILE)=0 THEN LET B(ZEIL
E)=B(ZEILE-1)
 2320 IF C(ZEILE)=0 THEN LET C(ZEIL
E)=C(ZEILE-1)
 2330 IF SPIELERZA=3 THEN GOTO 2370

 2340 IF D(ZEILE)=0 THEN LET D(ZEIL
E)=D(ZEILE-1)
 2350 IF SPIELERZA=4 THEN GOSUB 447
0
 2360 !
 2370 ! **************
 2380 ! * AUFBAU S$ *
 2390 ! **************
 2400 !
 2410 IF RTRIM$(UCASE$(D$))="J" THE
N
 2420   LET D$="H"
 2430 ELSE
 2440   LET D$=""
 2450 END IF
 2460 !
 2470 IF RTRIM$(UCASE$(E$))="J" THE
N
 2480   LET E$="S"
 2490 ELSE
 2500   LET E$=""
 2510 END IF
 2520 !
 2530 IF RTRIM$(UCASE$(F$))="J" THE
N
 2540   LET F$="S"
 2550 ELSE
 2560   LET F$=""
 2570 END IF
 2580 !
 2590 IF RTRIM$(UCASE$(G$))="J" THE
N
 2600   LET G$="C"
 2610 ELSE
 2620   LET G$=""
 2630 END IF
 2640 !
 2650 IF RTRIM$(UCASE$(H$))="J" THE
N
 2660   LET H$="R"
 2670 ELSE
 2680   LET H$=""
 2690 END IF
 2700 !
 2710 IF RTRIM$(UCASE$(W$))="J" THE
N
 2720   LET W$="B"
 2730 ELSE
 2740   LET W$=""
 2750 END IF
 2760 !
 2770 IF RTRIM$(UCASE$(S$))="J" THE
N
 2780   LET S$="Z"
 2790 ELSE
 2800   LET S$=""
 2810 END IF
 2820 !
 2830 FOR O=1 TO 9
 2840   IF O=1 THEN LET L$=RTRIM$(S
PART$)
 2850   IF O=2 THEN LET L$=RTRIM$(M
$)
 2860   IF O=3 THEN LET L$=RTRIM$(D
$)
 2870   IF O=4 THEN LET L$=RTRIM$(E
$)
 2880   IF O=5 THEN LET L$=RTRIM$(F
$)
 2890   IF O=6 THEN LET L$=RTRIM$(G
$)
 2900   IF O=7 THEN LET L$=RTRIM$(H
$)
 2910   IF O=8 THEN LET L$=RTRIM$(W
$)
 2920   IF O=9 THEN LET L$=RTRIM$(S
$)
 2930   LET SPANZEI$(ZEILE,O)=RTRIM
$(L$)
 2940 NEXT O
 2950 !
 2960 ! ********************
 2970 ! * TABELLENAUFBAU *
 2980 ! ********************
 2990 !
 3000 GOSUB 3580
 3010 IF RTRIM$(UCASE$(K$))<>"J" TH
EN
 3020   PRINT AT 21,1:"Spiel wurde
verloren! Spielwert:";
 3030   IF V1=0 THEN
 3040     PRINT SPWERT*2
 3050   ELSE
 3060     PRINT SPWERT*4
 3070   END IF
 3080 END IF
 3090 !
 3100 IF RTRIM$(UCASE$(K$))="J" THE
N
 3110   PRINT AT 21,1:"Spiel wurde
gewonnen! Spielwert:";
 3120   IF V1=0 THEN
 3130     PRINT SPWERT
 3140   ELSE
 3150     PRINT SPWERT*2
```

```
 3160    END IF
 3170  END IF
 3180  !
 3190  IF ZEILE>35 THEN
 3200    GOTO 4090
 3210  ELSE
 3220    IF ZEILE>18 THEN GOTO 3930
 3230  END IF
 3240  !
 3250  IF SPIELERZA=4 THEN GOTO 4370

 3260 PRINT AT ZEILE+1,1:ZEILE;TAB(
15);A(ZEILE);TAB(25);B(ZEILE);TAB(3
5);C(ZEILE)
 3270 LET K=45
 3280 FOR O=1 TO 9
 3290    PRINT AT ZEILE+1,K:SPANZEI$
(ZEILE,O)
 3300    LET K=K+1 ! GOSUB 2000
 3310 NEXT O
 3320 DO
 3330    LET ANTWORT$=INKEY$
 3340 LOOP UNTIL ANTWORT$<>""
 3350 LET ZEILE=ZEILE+1
 3360 !
 3370 ! ****************************

 3380 ! * LOESCHUNG VON VARIABELN *

 3390 ! ****************************
 3400 !
 3410 LET M$=""
 3420 LET D$=""
 3430 LET E$=""
 3440 LET F$=""
 3450 LET G$=""
 3460 LET H$=""
 3470 LET SPART$=""
 3480 LET SPWERT=0
 3490 LET SPIELER$=""
 3500 LET W$=""
 3510 !
 3520 IF RTRIM$(UCASE$(I$))="J" THE
N LET V=0
 3530 GOSUB 3630
 3540 GOTO 670
 3550 TEXT 80
 3560 END
 3570 !
 3580 ! ****************************
**
 3590 ! * LOESCHEN DER KOMANDOZEILE
N*
 3600 ! ****************************
**
 3610 !
 3620 PRINT AT 20,1:;:CALL SPC(60)
 3630 PRINT AT 21,1:;:CALL SPC(60)
 3640 RETURN
 3650 !
 3660 ! ****************************

 3670 ! * DIALOG BEI NULL-SPIELEN *

 3680 ! ****************************

 3690 !
 3700 DO
 3710    LET ANTWORT$=INKEY$
 3720 LOOP UNTIL ANTWORT$=""
 3730 GOSUB 3630
 3740 INPUT AT 21,1,PROMPT "Wurde c
ontra gegeben (j/n) :":G$
 3750 GOSUB 3630
 3760 !
 3770 INPUT AT 21,1,PROMPT "Wurde r
e gegeben (j/n) :":H$
 3780 GOSUB 3630
 3790 !
 3800 INPUT AT 21,1,PROMPT "Wurde B
ock gegeben (j/n) :":W$
 3810 GOSUB 3630
 3820 !
 3830 INPUT AT 21,1,PROMPT "Wurde Z
ippe gegeben (j/n) :":S$
 3840 GOSUB 3630
 3850 !
 3860 INPUT AT 21,1,PROMPT "Hat der
 Spielende gewonnen (j/n) :":K$
 3870 GOSUB 3630
 3880 !
 3890 INPUT AT 21,1,PROMPT "Haben S
ie einen Fehler gemacht (j/n) :":I$
 3900 IF RTRIM$(UCASE$(I$))="J" THE
N GOTO 1580
 3910 GOTO 1720
 3920 !
 3930 ! ****************************
****************
 3940 ! * TABELLENAUFBAU bei mehr a
ls 17 SPIELEN *
 3950 ! ****************************
****************
 3960 !
 3970 PRINT AT ZEILE-17,1:;:CALL SP
C(60)
 3980 PRINT AT ZEILE-16,1:;:CALL SP
C(60)
 3990 PRINT AT ZEILE-15,1:;:CALL SP
C(60)
 4000 PRINT AT ZEILE-14,1:;:CALL SP
C(60)
 4010 PRINT AT ZEILE-17,1:ZEILE;TAB
(15);A(ZEILE);TAB(25);B(ZEILE);TAB(
```

```
35);C(ZEILE)
 4020 LET K=45
 4030 FOR O=1 TO 9
 4040   PRINT AT ZEILE-17,K:SPANZEI
$(ZEILE,O)
 4050   LET K=K+1
 4060 NEXT O
 4070 GOTO 3320
 4080 !
 4090 ! ******************************
**************
 4100 ! * TABELLENAUFBAU bei mehr a
ls 35 SPIELEN *
 4110 ! ******************************
**************
 4120 !
 4130 PRINT AT ZEILE-35,1:;:CALL SP
C(60)
 4140 PRINT AT ZEILE-34,1:;:CALL SP
C(60)
 4150 PRINT AT ZEILE-33,1:;:CALL SP
C(60)
 4160 PRINT AT ZEILE-32,1:;:CALL SP
C(60)
 4170 PRINT AT ZEILE-35,1:ZEILE;TAB
(15);A(ZEILE);TAB(25);B(ZEILE);TAB(
35);C(ZEILE)
 4180 LET K=32
 4190 FOR O=1 TO 9
 4200   PRINT AT ZEILE-35,K:SPANZEI
$(ZEILE,O)
 4210   LET K=K+1
 4220 NEXT O
 4230 GOTO 3320
 4240 !
 4250 ! ******************************
****
 4260 ! * GEBERANZEIGE BEI 4 SPIELE
RN *
 4270 ! ******************************
****
 4280 !
 4290 IF GEBER=1 THEN PRINT AT ZEIL
E+1,5:"---"
 4300 IF GEBER=2 THEN PRINT AT ZEIL
E+1,12:"---"
 4310 IF GEBER=3 THEN PRINT AT ZEIL
E+1,19:"---"
 4320 IF GEBER=4 THEN PRINT AT ZEIL
E+1,26:"---"
 4330 LET GEBER=GEBER+1
 4340 IF GEBER=5 THEN LET GEBER=1
 4350 GOTO 990
 4360 !
 4370 ! ******************************
******
 4380 ! * TABELLENAUFBAU bei 4 SPIE
LERN *
 4390 ! ******************************
******
 4400 !
 4410 IF GEBER=2 THEN PRINT AT ZEIL
E+1,1:ZEILE;TAB(16);"---";TAB(22);B
(ZEILE);TAB(29);C(ZEILE);TAB(36);D(
ZEILE)
 4420 IF GEBER=3 THEN PRINT AT ZEIL
E+1,1:ZEILE;TAB(15);A(ZEILE);TAB(23
);"---";TAB(29);C(ZEILE);TAB(36);D(
ZEILE)
 4430 IF GEBER=4 THEN PRINT AT ZEIL
E+1,1:ZEILE;TAB(15);A(ZEILE);TAB(22
);B(ZEILE);TAB(30);"---";TAB(36);D(
ZEILE)
 4440 IF GEBER=1 THEN PRINT AT ZEIL
E+1,1:ZEILE;TAB(15);A(ZEILE);TAB(22
);B(ZEILE);TAB(29);C(ZEILE);TAB(37)
;"---"
 4450 GOTO 3270
 4460 !
 4470 ! ******************************
```

WENN IHR LISTING aus einer Fotokopie besteht, läuft dieses Programm beim Abtippen nicht!

```
*
 4480 ! * AUSGLEICH FUER DEN GEBER
*
 4490 ! ******************************
*
 4500 !
 4510 IF GEBER=2 THEN LET A(ZEILE)=
A(ZEILE-1)
 4520 IF GEBER=3 THEN LET B(ZEILE)=
B(ZEILE-1)
 4530 IF GEBER=4 THEN LET C(ZEILE)=
C(ZEILE-1)
 4540 IF GEBER=1 THEN LET D(ZEILE)=
D(ZEILE-1)
 4550 RETURN
 4560 !
 4570 ! ************************
 4580 ! * Mache Zeichenkette *
 4590 ! ************************
 4600 !
 4610 DEF SPC(LENGTH)
 4620   NUMERIC ZAEHLER
 4630   !
 4640   FOR ZAEHLER=1 TO LENGTH
 4650     PRINT CHR$(32);
 4660   NEXT ZAEHLER
 4670   !
 4680 END DEF
```

Der Rote Eric

Programmlänge: 25200 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128/Floppy

Ob man es nun glauben will oder nicht: Die Adventure-Spiele gehören zu den ältesten Spielprogrammen, die je auf Computern realisiert worden sind.

Natürlich waren diese Computer aus der Frühzeit des Elektronik-Zeitalters keine Home-Computer. Man fand sie vorwiegend in Universitäten oder großen Firmen, deren Finanzlage einen derartigen Luxus zuließ. Diese elektronischen Dinosaurier benötigten nämlich ein gewaltiges Heer an Ingenieuren und Wartungstechnikern, die durchgebrannte Röhren oder defekte Schaltrelais suchen und schließlich austauschen mußten.

Typisch für diese Zeit war, daß man Computer eher reparieren, denn programmieren mußte. Doch schaffte man endlich auch dieses Hindernis beiseite und begann mit der eigentlichen Programmierarbeit.
Natürlich waren die damaligen professionellen Programmierer genauso verspielt wie die heutigen 'Amateure'. Also nutzte man die Freizeit nach Feierabend, um kreativ zu werden. Neben anspruchsvolleren Simulationen (wie 'LIFE' in diesem Special) programmierte man auch Spiele. Da aber die meisten Terminals aus Kostengründen nur für die Wiedergabe von Text konzipiert waren, verboten sich schnelle Action-Spiele. Die Lösung waren intelligente Textabenteuer (eben die Adventures), die den Spieler in eine sagenhafte Welt entführten und ihn je nach Fantasie des Programmierers Ritter, Krieger oder gar Raumfahrer werden ließ.
Einer der Pioniere auf diesem Gebiet ist Scott Adams, dessen Abenteuerspiele zu den unbestrittenen Klassikern gehören. Bekannte Titel von ihm sind zum Beispiel 'Pirate Cove' und 'The incredible Hulk', wobei letzterer auch mit Grafik arbeitet und damit nicht mehr zu den Nur-Text-Spielen gehört. Obwohl der unerfahrene Leser bestimmt den mit aufwendiger Grafik verschönerten Adventures den Vorzug geben würde, zeigt sich nach dem Studium der Software-Top-Ten-Listen, daß die Textadventures bei weitem die erfolgreichsten sind. Und dieses Ergebnis kommt nicht von ungefähr: Schon die Infocom-Programmierer haben festgestellt, daß gute Beschreibungen der Handlungsorte und der Charaktere, die in dem Spiel vorkommen, weitaus aussagekräftiger und faszinierender sind als es Grafiken auf Home-Computern (beim heutigen Stand der Technik) je sein könnten. Das Softwarehaus Infocom stammt, wie übrigens auch die Computersprache LOGO, vom amerikanischen Massachusetts Institute of Technology (kurz 'MIT'). Blank und Lebling waren zwei der Gründungsmitglieder von Infocom, welches viele erfolgreiche Spiele produzierte. Etwa zum Beispiel 'Deadline', 'Wittness', 'Starcross', 'Planetfall' und 'Suspended', um nur einige zu nennen. Eines der berühmtesten und vielleicht sogar besten ist jedoch die Zork-Trilogie mit 'The Great Underground Empire', 'The Wizard of Frobozz' und 'The Dungeon Master', welche man sich unbedingt auf einem Commodore 64 oder Apple II ansehen sollte. Ein Geheimnis dieses unglaublichen Erfolges sind die ausgeklügelten Kommandosprachen, die die Verbindung des Spielers zur Abenteuerwelt herstellen. Infocom benutzt etwa 600 Worte in seiner Sprache 'Interlogic'. Es ist einsichtig, daß derart leistungsfähige Parser (so nennt man den Analyse-Teil der Kommandosprache) nur in Maschinensprache mit entsprechend großem Speicherplatz realisierbar sind. Trotzdem ist es uns gelungen, ein anspruchsvolles BASIC-Adventure auf dem Enterprise zu programmieren.

Spielbeschreibung:

'Der Rote Eric' gehört zur beliebten Klasse der Text-Adventures. Ziel des Spiels ist es, England von den Wikingern und besonders dem Roten Eric zu befreien. Um die gefangen gehaltene Prinzessin befreien zu können, muß man den Weg in die Burg des Wikinger-Häuptlings suchen und dort seinen mächtigsten Verbündeten, den bösen Zauberer, besiegen.
Die Kommandosprache ist dabei wegen des begrenzten BASIC-Speichers bewußt einfach gehalten worden und beschränkt sich auf einige wenige Standardworte.
Um den Reiz des Spiels nicht zu schmälern, soll an dieser Stelle nicht weiter auf den vorhandenen Wortschatz eingegangen werden. Einige Tips für den Anfang sind aber bestimmt ganz nützlich:
Verwendet werden sogenannte' Zwei-Wort-Kommandos (z.B. 'Schwert nehmen', 'Flasche öffnen', 'Eiche erklettern', 'Messer befestigen' usw.) und Richtungsbefehle, die meist nur aus einem Buchstaben bestehen (Nord, Süd, Ost, West, OBen, Unten). Um nach einem ungewollten Spielende (Tod der Spielfigur durch gefrässigen Drachen, o.ä.) nicht immer von vorne beginnen zu müssen, kann man den erreichten Spielstand auf Diskette abspeichern (Anpassung für Kassettenrekorder ist auch möglich). Dazu gibt man 'SVE' zum Speichern und 'LDE' zum Laden der Spieldaten zusammen mit dem dazugehörigen Filenamen ein, welcher frei gewählt werden kann.
Leider verarbeitet das IS-BASIC die deutschen Umlaute nicht korrekt, so daß man bei der Eingabe entsprechender Wörter die Umlaute in 'oe', 'ue' usw. umwandeln muß.

Variablenliste:

Im folgenden noch eine kurze Auflistung der verwendeten Variablennamen, um eventuelle Programmerweiterungen zu vereinfachen:

Numerische Variablen

AR - Anzahl der Räume (26)
AO - Anzahl der Objekte (41)
AF - Anzahl der Flags (6)
AV - Anzahl der Verben (15)
SP - Aktueller Raum des Spielers
LC - Licht (Gegenstände sichtbar oder nicht)
IX - maximal zu tragende Objektanzahl (6)
SR - Stärkepunkte (30)
ZF - Zufallsvariable
BM - Bemerkung ('tot' etc.)
MO - Aufenthaltsdauer bei bestimmten Gegenstand
ZR - weitere Zufallsvariable
GN - Anzahl der Bewegungen
P - Schleifenvariable
Q - Schleifenvariable
OZ - momentane Objekte
BN - Bonus
ZX - weitere Zufallsvariable

Stringvariablen

E$ - Eingabe
H$ - Subjekt-Objekt-Trennung
TW$ - Subjekt-Objekt-Trennung; Test auf Blank
WW$ - weitere Teileingabe ('Womit?')

Arrays

RA$(AR) - Raumbeschreibungen
OB$(AO) - Objektbeschreibungen
RN$(AO) - Bezeichnungen der Objekte
VE$(AV) - Verben
RI$(6) - Richtungen
OB(AO) - Standort der Objekte
IV(AO) - Inventory-Code (Inhaltsverzeichnis)
DE(AR,6) - Ausgänge aus dem jeweiligen Raum

Reiner Kunz

```
100 PROGRAM "Der-Rote-Eric"
105 !
110 TEXT 80
115 PRINT AT 5,10:"Einen Augenblick, bitte !!!"
120 PRINT AT 8,5:"Ich initialisie
```

```
re meinen Speicher ..."
  125 RANDOMIZE
  130 !
  135 ! ****************************
**********
  140 ! * Initialisierung der Syste
mvariablen *
  145 ! ****************************
**********
  150 !
  155 LET ANZRAEUME=26 !      Anzahl
 der Räume
  160 LET ANZOBJEKTE=41 !     Anzahl
 der Objekte
  165 LET ANZFLAGS=6 !        Anzahl
 der Flags
  170 LET ANZVERBEN=15 !      Anzahl
 der Verben
  175 LET SPIELERRAUM=1 !     Aktuel
ler Raum des Spielers
  180 LET LICHT=1 !           Licht
ein/aus
  185 LET MAXOBJEKT=6 !       maxima
l zu tragende Objektanzahl
  190 LET STAERKEPUNKTE=30 ! Stärke
punkte
  195 LET ZF=0 !              Zufall
svariable
  200 LET BEMERKUNG=0 !       Bemerk
ung ('tot', etc.)
  205 LET DAUER=0 !           Aufent
haltsdauer bei bestimmten Gegenstan
d
  210 LET ZR=0 !              zweite
 Zufallsvariable
  215 LET ANZBEWEGUNG=0 !     Anzahl
 der Bewegungen
  220 LET P=0 !               Schlei
fenvariable
  225 LET Q=0 !               Schlei
fenvariable
  230 LET ANZGEGENSTAND=0 !  wievie
l Gegenstände im Besitz ?
  235 LET BONUS=0 !           Bonus
  240 LET ZX=0 !              dritte
 Zufallsvariable
  245 !
  250 ! ****************************

  255 ! * Deklarierung der Arrays *

  260 ! ****************************

  265 !
  270 ! Raumbeschreibungen
  275 STRING RAUM$(ANZRAEUME)*40
  280 ! Objektbeschreibungen
  285 STRING OBJEKT$(ANZOBJEKTE)*30

  290 ! Bezeichnungen der Objekte
  295 STRING OBJNAMEN$(ANZOBJEKTE)*
4
  300 ! Verben
  305 STRING VERB$(ANZVERBEN)*3
  310 ! Richtungen
  315 STRING RICHTUNG$(6)*6
  320 NUMERIC DG(ANZRAEUME,6)
  325 ! Inhaltsverzeichnis
  330 NUMERIC INHALT(ANZOBJEKTE)
  335 ! Standort der Objekte
  340 NUMERIC ORTOBJEKT(ANZOBJEKTE)
  345 ! Flag-Array
  350 NUMERIC FLAG(ANZFLAGS)
  355 !
  360 ! ****************************
**********
  365 ! * Einlesen der Werte für di
e Arrays *
  370 ! ****************************
**********
  375 !
  380 FOR I=0 TO ANZFLAGS
  385   LET FLAG(I)=0
  390 NEXT I
  395 !
  400 RESTORE
  405 !
  410 FOR P=1 TO ANZRAEUME
  415   READ RAUM$(P)
  420   !
  425   FOR Q=1 TO 6
  430     READ DG(P,Q)
  435   NEXT Q
  440 NEXT P
  445 !
  450 FOR P=1 TO ANZOBJEKTE
  455   READ OBJEKT$(P),OBJNAMEN$(P
),ORTOBJEKT(P),INHALT(P)
  460 NEXT P
  465 !
  470 FOR P=1 TO ANZVERBEN
  475   READ VERB$(P)
  480 NEXT P
  485 !
  490 FOR P=1 TO 6
  495   READ RICHTUNG$(P)
  500 NEXT P
  505 !
  510 PRINT
  515 PRINT "Ich bin jetzt fertig !
!"
  520 PING
  525 PING
  530 !
```

```
  535 ! ****************************
**************************
  540 ! * DATA-Block für die zu ini
tialisierenden Arrays *
  545 ! ****************************
**************************
  550 !
  555 DATA "in einem Laubwald",4,0,
0,0,0,0
  560 DATA "in einem kleinen Baumha
us",0,0,0,0,0,1
  565 DATA "in einer verruchten Kne
ipe",6,0,0,0,0,0
  570 DATA "in einem Laubwald",0,0,
1,0,0,0
  575 DATA "auf dem Friedhof",0,6,0
,4,0,0
  580 DATA "auf der Dorfstrasse",9,
0,3,5,0,0
  585 DATA "in einer dunklen Höhle"
,0,8,0,18,0,0
  590 DATA "in einer dunklen Höhle"
,0,9,0,0,0,0
  595 DATA "am Ufer eines kleinen T
eiches",0,0,6,8,0,0
  600 DATA "in einer düsteren Dachk
ammer",0,0,0,0,0,11
  605 DATA "in einem Alchimistenlab
or",14,12,0,0,0,0
  610 DATA "in einer prunkvollen Sc
hlafkammer",0,0,0,11,0,0
  615 DATA "in der Gesindekammer",0
,14,0,0,0,0
  620 DATA "in einem bewachsenen In
nenhof",0,15,11,13,0,0
  625 DATA "im Garten",18,0,0,14,0,
0
  630 DATA "in einem Kaminzimmer",0
,17,0,0,0,0
  635 DATA "in der Diele",0,18,0,16
,0,0
  640 DATA "am verzauberten Tor",0,
7,0,0,0,0
  645 DATA "in einem Hundezwinger",
0,0,0,0,0,0
  650 DATA "in einem düsteren Verli
eß",22,0,0,0,0,0
  655 DATA "in einem Gang",23,0,19,
0,0,0
  660 DATA "in einer Wachenhöhle",2
4,0,0,0,0,0
  665 DATA "in einem Gang",0,0,21,0
,0,0
  670 DATA "in einer Folterkammer",
26,0,22,0,0,0
  675 DATA "in einem Schlangengrube
",0,0,23,0,0,0
  680 DATA "an einem unterirdischen
 Fluß",0,0,24,25,0,0
  685 !
  690 ! ****************************
**********
  695 ! * Zweiter DATA-Block für di
e Arrays *
  700 ! ****************************
**********
  705 !
  710 DATA "eine hohe Eiche","EICH"
,1,5
  715 DATA "einen Ast","AST ",1,0
  720 DATA "ein Messer","MESS",2,0
  725 DATA "ein Stück Schnur","SCHN
",2,0
  730 DATA "eine Schale","SCHA",3,0

  735 DATA "einen Speer","SPEE",0,0

  740 DATA "Giftkräuter","KRAE",4,0

  745 DATA "einen glatten Stein","S
TEI",5,0
  750 DATA "eine schmutzige Flasche
 ","FLAS",5,0
  755 DATA "*** GOLD ***","GOLD",7,
0
  760 DATA "einen großen Bären","BA
ER",8,1
  765 DATA "einen grossen Fisch","F
ISC",9,0
  770 DATA "einen Zauberring","RING
",0,0
  775 DATA "das Tor","TOR ",18,5
  780 DATA "einen kleinen Schlüssel
","SCHL",10,0
  785 DATA "ein altes Buch","BUCH",
11,0
  790 DATA "eine Flöte","FLOE",12,0

  795 DATA "einen Strick","STRI",14
,0
  800 DATA "eine Leiter","LEIT",15,
0
  805 DATA "einen Kamin","KAMI",16,
2
  810 DATA "einen verkohlten Zettel
","ZETT",0,0
  815 DATA "eine Öllampe","LAMP",17
,0
  820 DATA "Hunde","HUND",19,1
  825 DATA "*** Den roten Eric ***"
,"ERIC",23,3
  830 DATA "einen Zauberspruch","ZA
UB",0,2
  835 DATA "Ottern","OTTE",25,1
```

```
  840 DATA "ein Blatt Papier","PAPI
",25,0
  845 DATA "Wachen","WACH",22,1
  850 DATA "eine Streckbank","STRE"
,24,5
  855 DATA "=== Die Prinzessin ==="
,"PRIN",20,2
  860 DATA "einen feuerspeienden Dr
achen","DRAC",4,1
  865 DATA "eine alte Frau","FRAU",
13,2
  870 DATA "ein Gegengift","GEGE",0
,0
  875 DATA "eine Tarnkappe","KAPP",
0,0
  880 DATA "eine Grube","GRUB",1,2
  885 DATA "Zündhölzer","ZUEN",-1,0

  890 DATA "QQQQ","ORT ",0,1
  895 DATA "QQQQ","SIRU",0,1
  900 DATA "QQQQ","OEL ",0,1
  905 DATA "QQQQ","WASS",0,1
  910 DATA "eine Falltür","FALL",12
,5
  915 !
  920 ! ****************************
**********
  925 ! * Dritter DATA-Block für di
e Arrays *
  930 ! ****************************
**********
  935 !
  940 DATA "NEH","WEG","LES","BEN",
"UNT","ESS","TRI","BEF","ERK","OEF"
,"ANZ","DRE","SPR","TOE","KUE"
  945 !
  950 ! ****************************
**********
  955 ! * Vierter DATA-Block für di
e Arrays *
  960 ! ****************************
**********
  965 !
  970 DATA "NORDEN","OSTEN ","SUEDE
N","WESTEN","OBEN  ","UNTEN "
  975 !
  980 !
  985 SET #102:INK 2
  990 PRINT " STÜCKE:";ANZGEGENSTAN
D;" STÄRKE:";STAERKEPUNKTE;" BONUS:
";BONUS
  995 SET #102:INK 1
 1000 !
 1005 IF SPIELERRAUM>18 THEN LET LI
CHT=0
 1010 IF SPIELERRAUM<19 THEN LET LI
CHT=1
 1015 IF SPIELERRAUM>18 AND(ORTOBJE
KT(22)=SPIELERRAUM OR ORTOBJEKT(22)
=-1) AND(ORTOBJEKT(36)=SPIELERRAUM
OR ORTOBJEKT(36)=-1) THEN LET LICHT
=-1
 1020 !
 1025 IF LICHT=0 THEN
 1030   PRINT "-Es ist völlig dunke
l- Ich kann nichts mehr sehen!"
 1035   PRINT
 1040   GOTO 1255
 1045 END IF
 1050 !
 1055 PRINT "Ich bin ";RAUM$(SPIELE
RRAUM)
 1060 PRINT
 1065 PRINT "Ich sehe ";
 1070 !
 1075 FOR P=1 TO ANZOBJEKTE
 1080   IF ORTOBJEKT(P)=SPIELERRAUM
 THEN PRINT TAB(11);OBJEKT$(P)
 1085   IF ORTOBJEKT(P)=SPIELERRAUM
 THEN LET FF=1
 1090 NEXT P
 1095 !
 1100 IF FF=0 THEN PRINT "...gar ni
chts !"
 1105 LET FF=0
 1110 PRINT
 1115 PRINT "Ich kann nach ";
 1120 FOR P=1 TO 6
 1125   IF DG(SPIELERRAUM,P)<>0 THE
N PRINT TAB(15);RICHTUNG$(P)
 1130 NEXT P
 1135 !
 1140 IF SPIELERRAUM<19 THEN GOTO 1
210
 1145 LET ZF=RND(15)
 1150 IF ZF<>3 THEN GOTO 1210
 1155 !
 1160 PRINT "Die Wände stürzen ein.
 Ein Beben läßt den Boden erzittern
."
 1165 IF ST<10 THEN
 1170   LET BEMERKUNG=12
 1175   GOTO 4895
 1180 END IF
 1185 !
 1190 PRINT "Du wurdest eingeklemmt
, konntest dich aber befreien !"
 1195 LET ST=ST-6
 1200 GOTO 1255
 1205 !
 1210 FOR I=1 TO 40
 1215   PRINT "";
 1220 NEXT I
 1225 PRINT
```

```
 1230 !
 1235 ! ****************************
**************************
 1240 ! * PARSER-Routine zur Syntax
-Analyse der Eingaben *
 1245 ! ****************************
**************************
 1250 !
 1255 PRINT
 1260 PRINT "Was willst du tun..."
 1265 INPUT PROMPT ":":EINGABE$
 1270 LET EINGABE$=UCASE$(EINGABE$)

 1275 PRINT
 1280 !
 1285 IF EINGABE$="X" OR EINGABE$="
x" THEN PRINT "WIRKLICH (J/N) "
 1290 IF EINGABE$="X" OR EINGABE$="
x" THEN
 1295   DO
 1300     LET JN$=INKEY$
 1305   LOOP UNTIL JN$<>""
 1310   IF JN$="J" OR JN$="j" THEN
GOTO 5475: ! SPIELENDE
 1315   GOTO 985
 1320 END IF
 1325 !
 1330 IF EINGABE$="X" AND JN$="N" T
HEN GOTO 985
 1335 IF EINGABE$="X" OR EINGABE$="
x" THEN GOTO 1290
 1340 IF SPIELERRAUM=20 THEN GOTO 5
445
 1345 IF SPIELERRAUM=4 AND DG(4,2)=
0 THEN LET DAUER=DAUER+1
 1350 IF SPIELERRAUM=22 AND FLAG(4)
=0 THEN
 1355   LET WW=WW+1
 1360   IF WW=3 THEN
 1365     LET BEMERKUNG=3
 1370     GOTO 4895
 1375   END IF
 1380 END IF
 1385 !
 1390 IF STAERKEPUNKTE<=0 THEN LET
BEMERKUNG=4:GOTO 4895
 1395 !
 1400 IF SPIELERRAUM=19 THEN LET BE
MERKUNG=5:GOTO 4895
 1405 !
 1410 IF DAUER=5 THEN LET BEMERKUNG
=1:GOTO 4895
 1415 !
 1420 IF SPIELERRAUM=25 THEN
 1425   LET ZR=INT(RND(10)+1)
 1430   IF ZR<4 THEN
 1435     PRINT "Du wirst gebissen
!"
 1440     LET STAERKEPUNKTE=STAERKE
PUNKTE-5
 1445   END IF
 1450 END IF
 1455 !
 1460 IF LEN(EINGABE$)>2 THEN GOTO
1640
 1465 !
 1470 LET ANZBEWEGUNG=ANZBEWEGUNG+1

 1475 IF EINGABE$="N" AND DG(SPIELE
RRAUM,1)<>0 THEN
 1480   LET SPIELERRAUM=DG(SPIELERR
AUM,1)
 1485   GOTO 985
 1490 END IF
 1495 !
 1500 IF EINGABE$="O" AND DG(SPIELE
RRAUM,2)<>0 THEN
 1505   LET SPIELERRAUM=DG(SPIELERR
AUM,2)
 1510   GOTO 985
 1515 END IF
 1520 !
 1525 IF EINGABE$="S" AND DG(SPIELE
RRAUM,3)<>0 THEN
 1530   LET SPIELERRAUM=DG(SPIELERR
AUM,3)
 1535   GOTO 985
 1540 END IF
 1545 !
 1550 IF EINGABE$="W" AND DG(SPIELE
RRAUM,4)<>0 THEN
 1555   LET SPIELERRAUM=DG(SPIELERR
AUM,4)
 1560   GOTO 985
 1565 END IF
 1570 !
 1575 IF EINGABE$="OB" AND DG(SPIEL
ERRAUM,5)<>0 THEN
 1580   LET SPIELERRAUM=DG(SPIELERR
AUM,5)
 1585   GOTO 985
 1590 END IF
 1595 !
 1600 IF EINGABE$="U" AND DG(SPIELE
RRAUM,6)<>0 THEN
 1605   LET SPIELERRAUM=DG(SPIELERR
AUM,6)
 1610   GOTO 985
 1615 END IF
 1620 !
 1625 PRINT "Es führt kein Weg dort
hin."
 1630 GOTO 1235
 1635 !
```

```
 1640 IF LEN(EINGABE$)<>3 THEN GOTO
 1765
 1645 IF EINGABE$="SVE" THEN GOTO 5
015
 1650 IF EINGABE$="LDE" THEN GOTO 5
220
 1655 IF EINGABE$="ORT" THEN GOTO 9
85
 1660 IF EINGABE$="PKT" THEN
 1665   PRINT "Du hast im Moment ";
BONUS;" Punkte."
 1670   GOTO 1255
 1675 END IF
 1680 !
 1685 IF EINGABE$="STR" THEN
 1690   PRINT "Du hast ";STAERKEPUN
KTE;" Stärkepunkte"
 1695   GOTO 1255
 1700 END IF
 1705 !
 1710 IF EINGABE$<>"INV" THEN
 1715   PRINT "Wie bitte ?"
 1720   GOTO 1235
 1725 END IF
 1730 !
 1735 PRINT "Du hast in deinem Besi
tz :"
 1740 FOR P=1 TO ANZOBJEKTE
 1745   IF ORTOBJEKT(P)=-1 THEN PRI
NT OBJEKT$(P)
 1750 NEXT P
 1755 GOTO 1235
 1760 !
 1765 LET H$=EINGABE$(1:4)
 1770 FOR P=1 TO LEN(EINGABE$)
 1775   IF EINGABE$(P:P)=" " THEN
 1780     LET TW$=EINGABE$(P+1:(P+1
)+2)
 1785     GOTO 1810
 1790   END IF
 1795 NEXT P
 1800 PRINT "FALSCHER SATZBAU":GOTO
 1255
 1805 !
 1810 FOR HS=1 TO ANZOBJEKTE
 1815   IF OBJNAMEN$(HS)=H$ THEN GO
TO 1840
 1820 NEXT HS
 1825 IF BEMERKUNG=12 THEN PRINT "D
ir ist ein Stein auf den Kopf gefal
len"
 1830 PRINT "ICH VERSTEHE DAS OBJEK
T NICHT":GOTO 1255
 1835 !
 1840 FOR VS=1 TO ANZVERBEN
 1845   IF VERB$(VS)=TW$ THEN GOTO
1870
 1850 NEXT VS
 1855 !
 1860 PRINT "ICH VERSTEHE DAS VERB
NICHT":GOTO 1255
 1865 !
 1870 ON VS GOTO 1890,2195,2385,255
5,2845,3610,3730,3870,4010,4090,433
0,4370,4470
 1875 LET VS=VS-13
 1880 ON VS GOTO 4570,4805
 1885 !
 1890 LET ANZGEGENSTAND=0
 1895 FOR Q=1 TO ANZOBJEKTE
 1900   IF ORTOBJEKT(Q)=-1 THEN LET
 ANZGEGENSTAND=ANZGEGENSTAND+1
 1905 NEXT Q
 1910 !
 1915 IF ORTOBJEKT(HS)=-1 THEN
 1920   PRINT "Das habe ich schon b
ei mir."
 1925   GOTO 1255
 1930 END IF
 1935 !
 1940 IF ORTOBJEKT(HS)<>SPIELERRAUM
 THEN
 1945   PRINT "Das kann ich hier ni
cht sehen."
 1950   GOTO 1255
 1955 END IF
 1960 !
 1965 IF HS=7 AND SPIELERRAUM=4 AND
 ORTOBJEKT(31)<>99 THEN
 1970   LET BEMERKUNG=13
 1975   GOTO 4895
 1980 END IF
 1985 !
 1990 IF ANZGEGENSTAND>=MAXOBJEKT T
HEN
 1995   PRINT "Soviel kann ich nich
t tragen."
 2000   GOTO 1255
 2005 END IF
 2010 !
 2015 IF HS=5 THEN GOTO 2935
 2020 IF INHALT(HS)=5 THEN
 2025   PRINT "Ich bin nicht stark
genug."
 2030   GOTO 1255
 2035 END IF
 2040 !
 2045 IF INHALT(HS)=4 THEN
 2050   PRINT "OHNE BEHÄLTER ?"
 2055   GOTO 1255
 2060 END IF
 2065 !
 2070 IF INHALT(HS)=3 THEN
 2075   LET BEMERKUNG=6
```

```
 2080   GOTO 4895
 2085 END IF
 2090 !
 2095 IF INHALT(HS)=2 THEN
 2100   PRINT "Tut mir leid. Das ge
ht nicht!"
 2105   GOTO 1255
 2110 END IF
 2115 !
 2120 IF INHALT(HS)=1 THEN
 2125   PRINT "VORSICHT. Das ist zu
 gefährlich!"
 2130   GOTO 1255
 2135 END IF
 2140 !
 2145 IF HS=21 THEN GOTO 2510
 2150 IF HS=8 THEN GOTO 3560
 2155 LET ORTOBJEKT(HS)=-1:GOTO 985
 2160 IF HS=13 THEN LET BONUS=BONUS
+20
 2165 IF HS=15 THEN LET BONUS=BONUS
+25
 2170 !
 2175 IF HS=33 THEN LET BONUS=BONUS
+15
 2180 IF HS=34 THEN LET BONUS=BONUS
+20
 2185 GOTO 1255
 2190 !
 2195 IF ORTOBJEKT(HS)<>-1 THEN
 2200   PRINT "Das hast du nicht be
i dir."
 2205   GOTO 1255
 2210 END IF
 2215 !
 2220 IF SPIELERRAUM=8 AND HS=9 THE
N
 2225   PRINT "Dem Bären schmeckt e
s."
 2230   LET ORTOBJEKT(9)=99
 2235   LET ANZGEGENSTAND=ANZGEGENS
TAND-1
 2240   GOTO 1255
 2245 END IF
 2250 !
 2255 IF SPIELERRAUM=8 AND HS=7 AND
 ORTOBJEKT(11)<>99 THEN PRINT "Der
Bär frißt es und fällt um."
 2260 IF SPIELERRAUM=8 AND HS=7 AND
 ORTOBJEKT(11)<>99 THEN
 2265   LET BONUS=BONUS+20
 2270   LET DG(8,4)=7
 2275   LET ORTOBJEKT(11)=99
 2280   LET ANZGEGENSTAND=ANZGEGENS
TAND-1
 2285 END IF
 2290 !
 2295 IF SPIELERRAUM=8 AND HS=7 AND
 ORTOBJEKT(11)=99 THEN
 2300   LET ORTOBJEKT(7)=99
 2305   GOTO 985
 2310 END IF
 2315 !
 2320 IF SPIELERRAUM=13 AND HS=10 T
HEN PRINT "Sie ist dankbar und sche
nkt dir ein wirksames";
 2325 IF SPIELERRAUM=13 AND HS=10 T
HEN
 2330   PRINT " GEGENGIFT."
 2335   LET ORTOBJEKT(33)=-1
 2340   LET ORTOBJEKT(10)=99
 2345   GOTO 1255
 2350 END IF
 2355 !
 2360 LET ORTOBJEKT(HS)=SPIELERRAUM

 2365 PRINT "O.K."
 2370 LET ANZGEGENSTAND=ANZGEGENSTA
ND-1
 2375 GOTO 1255
 2380 !
 2385 IF (ORTOBJEKT(27)=SPIELERRAUM
 OR ORTOBJEKT(27)=-1) AND HS=27 THE
N
 2390   PRINT "Darauf steht:"
 2395   PRINT "'NACH NORD"
 2400 END IF
 2405 !
 2410 IF (ORTOBJEKT(27)=SPIELERRAUM
 OR ORTOBJEKT(27)=-1) AND HS=27 THE
N
 2415   PRINT " SÜDEN,OSTEN,WESTEN,
ENTKOMMST DU ";
 2420 END IF
 2425 !
 2430 IF (ORTOBJEKT(27)=SPIELERRAUM
 OR ORTOBJEKT(27)=-1) AND HS=27 THE
N
 2435   PRINT " JEDEM RAUM AM BESTE
N'"
 2440   GOTO 1255
 2445 END IF
 2450 !
 2455 IF (ORTOBJEKT(16)=-1 OR ORTOB
JEKT(16)=SPIELERRAUM) AND HS=16 THE
N PRINT "Es enthält einen";
 2460 IF (ORTOBJEKT(16)=-1 OR ORTOB
JEKT(16)=SPIELERRAUM) AND HS=16 THE
N
 2465   PRINT " ZAUBERSPRUCH..."
 2470   LET BONUS=BONUS+10
 2475 END IF
 2480 !
 2485 IF (ORTOBJEKT(16)=-1 OR ORTOB
```

```
JEKT(16)=SPIELERRAUM) AND HS=16 THE
N
 2490   LET FLAG(4)=1
 2495   GOTO 1255
 2500 END IF
 2505 !
 2510 IF HS=21 AND(ORTOBJEKT(21)=-1
 OR ORTOBJEKT(21)=SPIELERRAUM) THEN
 PRINT "Das Papier ist mit einer se
ltsamen";
 2515 IF HS=21 AND(ORTOBJEKT(21)=-1
 OR ORTOBJEKT(21)=SPIELERRAUM) THEN

 2520   PRINT " Flüssigkeit benetzt
."
 2525   LET STAERKEPUNKTE=STAERKEPU
NKTE-3
 2530 END IF
 2535 !
 2540 IF HS=21 AND(ORTOBJEKT(21)=-1
 OR ORTOBJEKT(21)=SPIELERRAUM) THEN
 GOTO 1255
 2545 !
 2550 PRINT "Ich finde hier nichts
zum lesen.":GOTO 1255
 2555 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) THEN

 2560   PRINT "WIE DENN ?"
 2565   GOTO 1255
 2570 END IF
 2575 !
 2580 IF HS=17 AND(ORTOBJEKT(17)=SP
IELERRAUM OR ORTOBJEKT(17)=-1) AND
SPIELERRAUM=25 THEN PRINT "Die Schl
angen sind";
 2585 IF HS=17 AND(ORTOBJEKT(17)=SP
IELERRAUM OR ORTOBJEKT(17)=-1) AND
SPIELERRAUM=25 THEN
 2590   PRINT " in Trance."
 2595   LET DG(25,2)=26
 2600 END IF
 2605 !
 2610 IF HS=17 AND(ORTOBJEKT(17)=SP
IELERRAUM OR ORTOBJEKT(17)=-1) AND
SPIELERRAUM=25 THEN
 2615   LET BONUS=BONUS+20
 2620   GOTO 1255
 2625 END IF
 2630 !
 2635 IF HS=17 AND(ORTOBJEKT(17)=SP
IELERRAUM OR ORTOBJEKT(17)=-1) THEN

 2640   PRINT "Hört sich gut an!"
 2645   GOTO 1255
 2650 END IF
 2655 !
 2660 IF HS=19 AND(ORTOBJEKT(19)=SP
IELERRAUM OR ORTOBJEKT(19)=-1) AND
SPIELERRAUM=11 THEN
 2665   LET ORTOBJEKT(19)=11
 2670   LET INHALT(19)=3
 2675 END IF
 2680 !
 2685 IF HS=19 AND(ORTOBJEKT(19)=SP
IELERRAUM OR ORTOBJEKT(19)=-1) AND
SPIELERRAUM=11 THEN
 2690   PRINT "O.K."
 2695   LET DG(11,5)=10
 2700   GOTO 1255
 2705 END IF
 2710 !
 2715 IF HS=29 AND SPIELERRAUM=24 T
HEN
 2720   PRINT "Das war ein Fehler -
 Dich schmerzt alles."
 2725   LET STAERKEPUNKTE=STAERKEPU
NKTE-5
 2730   GOTO 1255
 2735 END IF
 2740 !
 2745 IF HS<>34 OR(ORTOBJEKT(3)<>SP
IELERRAUM AND ORTOBJEKT(34)<>-1) TH
EN GOTO 2785
 2750 !
 2755 IF SPIELERRAUM=22 THEN
 2760   PRINT "O.K."
 2765   LET DG(22,3)=20
 2770   LET BONUS=BONUS+15
 2775 END IF
 2780 !
 2785 PRINT "Sie verschwindet."
 2790 LET ORTOBJEKT(34)=99
 2795 GOTO 1255
 2800 !
 2805 IF HS=18 AND(ORTOBJEKT(18)=SP
IELERRAUM OR ORTOBJEKT(18)=-1) THEN

 2810   LET BEMERKUNG=7
 2815   GOTO 4895
 2820 END IF
 2825 !
 2830 PRINT "VERSUCH'S ANDERS!"
 2835 GOTO 1255
 2840 !
 2845 IF HS=35 AND SPIELERRAUM=1 AN
D ORTOBJEKT(34)=0 THEN
 2850   LET ORTOBJEKT(34)=1
 2855   GOTO 985
 2860 END IF
 2865 !
 2870 IF HS=9 AND(ORTOBJEKT(9)=SPIE
LERRAUM OR ORTOBJEKT(9)=-1) THEN
 2875   PRINT "Sie enthält Sirup."
```

```
 2880   GOTO 1255
 2885 END IF
 2890 !
 2895 IF HS=12 AND(ORTOBJEKT(12)=SP
IELERRAUM OR ORTOBJEKT(12)=-1) THEN
 PRINT "Er scheint etwas verschluck
t zu haben."
 2900 IF HS=12 AND(ORTOBJEKT(12)=SP
IELERRAUM OR ORTOBJEKT(12)=-1) THEN
 GOTO 1255
 2905 !
 2910 IF HS=14 AND SPIELERRAUM=18 T
HEN
 2915   PRINT "Es ist alt und rosti
g."
 2920   GOTO 1255
 2925 END IF
 2930 !
 2935 IF HS=5 AND(ORTOBJEKT(5)=SPIE
LERRAUM OR ORTOBJEKT(5)=-1) THEN PR
INT "Der Wirt kommt und wirft dich"
;
 2940 IF HS=5 AND(ORTOBJEKT(5)=SPIE
LERRAUM OR ORTOBJEKT(5)=-1) THEN
 2945   PRINT " hinaus ! (AUTSCH !)
"
 2950   LET STAERKEPUNKTE=STAERKEPU
NKTE-4
 2955   LET SPIELERRAUM=6
 2960 END IF
 2965 IF HS=5 AND ORTOBJEKT(5)=3 AN
D SPIELERRAUM=6 THEN
 2970   FOR XX=1 TO 1000
 2975   NEXT XX
 2980   GOTO 985
 2985 END IF
 2990 !
 2995 IF HS=16 AND(ORTOBJEKT(16)=SP
IELERRAUM OR ORTOBJEKT(16)=-1) THEN

 3000   PRINT "Es ist ein altes Zau
berbuch."
 3005   GOTO 1255
 3010 END IF
 3015 !
 3020 IF HS=31 AND SPIELERRAUM=4 AN
D ORTOBJEKT(31)<>99 THEN
 3025   LET BEMERKUNG=8
 3030   GOTO 4895
 3035 END IF
 3040 !
 3045 IF HS=31 AND SPIELERRAUM=4 TH
EN
 3050   PRINT "Er ist über dreißig
Fuß lang!"
 3055   GOTO 1255
 3060 END IF
 3065 !
 3070 IF HS=11 AND SPIELERRAUM=8 AN
D ORTOBJEKT(11)<>99 THEN
 3075   LET BEMERKUNG=8
 3080   GOTO 4895
 3085 END IF
 3090 !
 3095 IF HS=11 AND SPIELERRAUM=8 TH
EN PRINT "Sein Fell riecht etwas sc
hlecht...";
 3100 IF HS=11 AND SPIELERRAUM=8 TH
EN
 3105   PRINT "Dir wird übel..."
 3110   LET STAERKEPUNKTE=STAERKEPU
NKTE-2
 3115   GOTO 3530
 3120 END IF
 3125 !
 3130 IF HS=29 AND SPIELERRAUM=24 T
HEN
 3135   PRINT "Du hast dir die Nase
 eingeklemmt."
 3140   LET STAERKEPUNKTE=STAERKEPU
NKTE-2
 3145   GOTO 1255
 3150 END IF
 3155 !
 3160 IF HS=1 AND SPIELERRAUM=1 THE
N
 3165   PRINT "Im Baumwipfel ist ei
n Baumhaus."
 3170   GOTO 1255
 3175 END IF
 3180 !
 3185 IF HS=2 AND(ORTOBJEKT(2)=SPIE
LERRAUM OR ORTOBJEKT(2)=-1) THEN
 3190   PRINT "Er ist fest und lang
."
 3195   GOTO 1255
 3200 END IF
 3205 !
 3210 IF HS=37 AND SPIELERRAUM=11 A
ND DG(11,5)=0 THEN
 3215   PRINT "An der Decke ist ein
e Falltür."
 3220   GOTO 1255
 3225 END IF
 3230 !
 3235 IF HS=3 AND(ORTOBJEKT(3)=SPIE
LERRAUM OR ORTOBJEKT(3)=-1) THEN
 3240   PRINT "Es ist sehr schartig
."
 3245   GOTO 1255
 3250 END IF
 3255 !
 3260 IF HS=6 AND(ORTOBJEKT(6)=SPIE
LERRAUM OR ORTOBJEKT(6)=-1) THEN
```

```
 3265   PRINT "Er sieht sehr robust
 aus."
 3270   GOTO 1255
 3275 END IF
 3280 !
 3285 IF HS=7 AND(ORTOBJEKT(7)=SPIE
LERRAUM OR ORTOBJEKT(7)=-1) THEN PR
INT "Ein süßlicher Duft steigt dir"
;
 3290 IF HS=7 AND(ORTOBJEKT(7)=SPIE
LERRAUM OR ORTOBJEKT(7)=-1) THEN PR
INT "in die Nase. Die wird schwindl
ig."
 3295 IF HS=7 AND(ORTOBJEKT(7)=SPIE
LERRAUM OR ORTOBJEKT(7)=-1) THEN
 3300   LET STAERKEPUNKTE=STAERKEPU
NKTE-3
 3305   GOTO 1255
 3310 END IF
 3315 !
 3320 IF HS=15 AND(ORTOBJEKT(15)=SP
IELERRAUM OR ORTOBJEKT(15)=-1) THEN
 PRINT "Er trägt ein Etikett mit de
r";
 3325 IF HS=15 AND(ORTOBJEKT(15)=SP
IELERRAUM OR ORTOBJEKT(15)=-1) THEN

 3330   PRINT "Aufschrift 'F.L..ÜR'
."
 3335   GOTO 1255
 3340 END IF
 3345 !
 3350 IF HS=20 AND SPIELERRAUM=16 A
ND ORTOBJEKT(21)=0 THEN
 3355   LET ORTOBJEKT(21)=16
 3360   PRINT "O.K."
 3365   GOTO 1255
 3370 END IF
 3375 !
 3380 IF HS=22 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN
 PRINT "Sie ist noch gut zu gebrauc
hen."
 3385 IF HS=22 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN
 GOTO 1255
 3390 !
 3395 IF HS=34 THEN GOTO 2745
 3400 !
 3405 IF HS=28 AND ORTOBJEKT(28)=22
 AND SPIELERRAUM=22 THEN
 3410   PRINT "Sie sehen gefährlich
 aus."
 3415   GOTO 1255
 3420 END IF
 3425 !
 3430 IF HS=18 AND(ORTOBJEKT(18)=SP
IELERRAUM OR ORTOBJEKT(18)=-1) THEN
 PRINT "OHA-Du hast dich darin";
 3435 IF HS=18 AND(ORTOBJEKT(18)=SP
IELERRAUM OR ORTOBJEKT(18)=-1) THEN

 3440   PRINT " verfangen."
 3445   LET STAERKEPUNKTE=STAERKEPU
NKTE-2
 3450   GOTO 1255
 3455 END IF
 3460 !
 3465 IF HS=19 AND(ORTOBJEKT(19)=SP
IELERRAUM OR ORTOBJEKT(19)=-1) THEN
 PRINT "Sie ist dir auf den Kopf ";
 3470 IF HS=19 AND(ORTOBJEKT(19)=SP
IELERRAUM OR ORTOBJEKT(19)=-1) THEN

 3475   PRINT " gefallen."
 3480   LET ST=ST-1
 3485   GOTO 1255
 3490 END IF
 3495 !
 3500 IF HS=41 AND SPIELERRAUM=12 T
HEN PRINT "Sie scheint in unterirdi
sche KATAKOMBEN zu";
 3505 IF HS=41 AND SPIELERRAUM=12 T
HEN
 3510   PRINT " führen."
 3515   GOTO 1255
 3520 END IF
 3525 !
 3530 IF HS=10 AND(ORTOBJEKT(10)=SP
IELERRAUM OR ORTOBJEKT(10)=-1) THEN
 PRINT "Es sind zwanzig (!)";
 3535 IF HS=10 AND(ORTOBJEKT(10)=SP
IELERRAUM OR ORTOBJEKT(10)=-1) THEN

 3540   PRINT " Golddublonen."
 3545   GOTO 1255
 3550 END IF
 3555 !
 3560 IF HS=8 AND ORTOBJEKT(8)=SPIE
LERRAUM THEN PRINT "Unter dem Stein
 sitzt eine giftige Kröte";
 3565 IF HS=8 AND ORTOBJEKT(8)=SPIE
LERRAUM THEN
 3570   PRINT " - sie beißt dich!"
 3575   LET STAERKEPUNKTE=STAERKEPU
NKTE-3
 3580   GOTO 1255
 3585 END IF
 3590 !
 3595 PRINT "Ich kann nichts besond
eres finden."
 3600 GOTO 1255
 3605 !
 3610 IF SPIELERRAUM=7 AND(ORTOBJEK
```

```
T(7)=SPIELERRAUM OR ORTOBJEKT(7)=-1
) THEN
 3615   LET BEMERKUNG=9
 3620   GOTO 4895
 3625 END IF
 3630 !
 3635 IF HS=38 AND(ORTOBJEKT(9)=SPI
ELERRAUM OR ORTOBJEKT(9)=-1) THEN
 3640   LET FLAG(2)=-1
 3645   PRINT "O.K."
 3650   GOTO 1255
 3655 END IF
 3660 !
 3665 IF HS=12 AND(ORTOBJEKT(12)=SP
IELERRAUM OR ORTOBJEKT(12)=-1) THEN

 3670   PRINT "Du beißt auf einen R
ing..."
 3675   LET STAERKEPUNKTE=STAERKEPU
NKTE+3
 3680 END IF
 3685 !
 3690 IF HS=12 AND(ORTOBJEKT(12)=SP
IELERRAUM OR ORTOBJEKT(12)=-1) THEN

 3695   LET ORTOBJEKT(13)=SPIELERRA
UM
 3700   LET ORTOBJEKT(12)=99
 3705   GOTO 1255
 3710 END IF
 3715 !
 3720 PRINT "Ich sehe hier nichts e
ssbares !":GOTO 1255
 3725 !
 3730 IF HS=39 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN

 3735   PRINT "PFUI TEUFEL!"
 3740   LET STAERKEPUNKTE=STAERKEPU
NKTE-2
 3745   GOTO 1255
 3750 END IF
 3755 !
 3760 IF HS<>33 THEN GOTO 3825
 3765 !
 3770 IF ORTOBJEKT(33)<>SPIELERRAUM
 AND ORTOBJEKT(33)<>-1 THEN GOTO 38
25
 3775 !
 3780 PRINT "Es geht dir sofort bes
ser!"
 3785 LET STAERKEPUNKTE=STAERKEPUNK
TE+8
 3790 LET BONUS=BONUS+10
 3795 GOTO 1255
 3800 !
 3805 PRINT "O.K."
 3810 LET ORTOBJEKT(33)=99
 3815 GOTO 1255
 3820 !
 3825 IF HS=40 AND SPIELERRAUM=26 T
HEN
 3830   PRINT "Es ist das 'Wasser d
er Kräfte'"
 3835   LET STAERKEPUNKTE=STAERKEPU
NKTE+5
 3840   GOTO 1255
 3845 END IF
 3850 !
 3855 PRINT "Ich sehe hier nichts z
um trinken!"
 3860 GOTO 1255
 3865 !
 3870 IF ORTOBJEKT(4)<>SPIELERRAUM
AND ORTOBJEKT(4)<>-1 THEN
 3875   PRINT "Du hast nichts zum b
efestigen!"
 3880   GOTO 1255
 3885 END IF
 3890 !
 3895 INPUT PROMPT "Woran ":WW$
 3900 LET WW$=UCASE$(WW$)
 3905 !
 3910 IF HS<>3 OR(ORTOBJEKT(3)<>SPI
ELERRAUM AND ORTOBJEKT(3)<>-1) OR W
W$<>"AST" OR(ORTOBJEKT(2)<>SPIELERR
AUM AND ORTOBJEKT(2)<>-1) THEN GOTO
 3945
 3915 PRINT "Du erhälst einen guten
 Speer!"
 3920 LET BONUS=BONUS+20
 3925 LET ORTOBJEKT(2)=99
 3930 LET ORTOBJEKT(3)=99
 3935 LET ORTOBJEKT(6)=-1
 3940 LET ORTOBJEKT(4)=99:GOTO 1255

 3945 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) THEN

 3950   PRINT "Er verschwindet!"
 3955   LET BONUS=BONUS-25
 3960   LET ORTOBJEKT(13)=99
 3965 END IF
 3970 !
 3975 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) THEN

 3980   PRINT
 3985   GOTO 1255
 3990 END IF
 3995 !
 4000 PRINT "Du bist zu ungeschickt
!":GOTO 1255
 4005 !
```

```
 4010 IF HS<>1 OR SPIELERRAUM<>1 TH
EN GOTO 4065
 4015 !
 4020 LET ZX=RND(10)+1
 4025 IF ZX<3 THEN
 4030   PRINT "Du bist abgerutscht!
"
 4035   LET STAERKEPUNKTE=STAERKEPU
NKTE-3
 4040   GOTO 1255
 4045 END IF
 4050 !
 4055 LET SPIELERRAUM=2:LET BONUS=B
ONUS+15:GOTO 985
 4060 !
 4065 IF HS=19 AND(ORTOBJEKT(19)=SP
IELERRAUM OR ORTOBJEKT(19)=-1) THEN
 PRINT "Du mußt sie erst irgendwo a
nlehnen!"
 4070 IF HS=19 AND(ORTOBJEKT(19)=SP
IELERRAUM OR ORTOBJEKT(19)=-1) THEN
 PRINT :GOTO 1255
 4075 !
 4080 PRINT "???!":GOTO 1255
 4085 !
 4090 IF HS=9 THEN GOTO 2870
 4095 !
 4100 IF HS=14 AND SPIELERRAUM=18 T
HEN
 4105   PRINT "Guter Witz!"
 4110   GOTO 1255
 4115 END IF
 4120 !
 4125 IF HS=22 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN

 4130   LET BONUS=BONUS-20
 4135   PRINT "DU PECHVOGEL...Die L
ampe ";
 4140 END IF
 4145 !
 4150 IF HS=22 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN

 4155   LET BONUS=BONUS-20
 4160   PRINT "ist zerbrochen und d
u ";
 4165 END IF
 4170 !
 4175 IF HS=22 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN

 4180   LET BONUS=BONUS-20
 4185   PRINT "bist auf dem Öl ausg
erutscht"
 4190 END IF
 4195 !
 4200 IF HS=22 AND(ORTOBJEKT(22)=SP
IELERRAUM OR ORTOBJEKT(22)=-1) THEN

 4205   LET BONUS=BONUS-20
 4210   LET ORTOBJEKT(22)=99
 4215   LET STAERKEPUNKTE=STAERKEPU
NKTE-2
 4220   GOTO 1255
 4225 END IF
 4230 !
 4235 IF HS=41 AND SPIELERRAUM=12 T
HEN
 4240   INPUT PROMPT "Womit ":WW$
 4245   LET WW$=UCASE$(WW$)
 4250 END IF
 4255 !
 4260 IF HS=41 AND SPIELERRAUM=12 T
HEN
 4265   IF WW$="SCHL" AND ORTOBJEKT
(15)=SPIELERRAUM OR ORTOBJEKT(15)=-
1 THEN
 4270     PRINT "O.K."
 4275     LET BONUS=BONUS+15
 4280   END IF
 4285 END IF
 4290 !
 4295 IF HS=41 AND SPIELERRAUM=12 T
HEN
 4300   LET DG(12,6)=21
 4305   GOTO 1255
 4310 END IF
 4315 !
 4320 PRINT "WAS ?":GOTO 1255
 4325 !
 4330 IF HS=22 THEN PRINT "O.K.":GO
TO 1255
 4335 !
 4340 PRINT "Du verbrauchst alle de
ine Streichhölzer, kannst es aber n
icht"
 4345 PRINT "anzünden."
 4350 LET ORTOBJEKT(36)=99
 4355 LET BONUS=BONUS+10
 4360 GOTO 1255
 4365 !
 4370 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) AND
SPIELERRAUM=18 THEN PRINT "Der Ring
 löst sich auf !"
 4375 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) AND
SPIELERRAUM=18 THEN
 4380   LET ORTOBJEKT(13)=99
 4385   LET DG(18,3)=15
 4390 END IF
 4395 !
 4400 IF HS=13 AND ORTOBJEKT(13)=99
```

```
 AND SPIELERRAUM=18 THEN
 4405   LET DG(18,4)=17
 4410   GOTO 985
 4415 END IF
 4420 !
 4425 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) THEN
 PRINT "Er läßt sich im Moment nich
t dre";
 4430 IF HS=13 AND(ORTOBJEKT(13)=SP
IELERRAUM OR ORTOBJEKT(13)=-1) THEN

 4435   PRINT "hen."
 4440   GOTO 1255
 4445 END IF
 4450 !
 4455 PRINT "O.K."
 4460 GOTO 1255
 4465 !
 4470 IF HS=25 AND FLAG(4)=1 AND SP
IELERRAUM=23 THEN
 4475   LET BONUS=BONUS+35
 4480   PRINT "Der Zauberer verschw
indet!"
 4485   LET ORTOBJEKT(24)=99
 4490 END IF
 4495 !
 4500 IF HS=25 AND FLAG(4)=1 AND SP
IELERRAUM=23 THEN
 4505   LET BONUS=BONUS+35
 4510   LET DG(23,1)=25
 4515   GOTO 1255
 4520 END IF
 4525 !
 4530 IF HS=25 AND FLAG(4)=1 THEN
 4535   LET BEMERKUNG=2
 4540   GOTO 4895
 4545 END IF
 4550 !
 4555 INPUT PROMPT "Sprich dich nur
 aus. Was willst du sagen ":SQ$
 4560 LET SQ$=UCASE$(SQ$)
 4565 !
 4570 IF SPIELERRAUM<>4 AND SPIELER
RAUM<>8 AND SPIELERRAUM<>25 AND SPI
ELERRAUM<>23 THEN
 4575   PRINT "Versuch es anders !"

 4580   GOTO 1255
 4585 END IF
 4590 !
 4595 INPUT PROMPT "Mit welcher Waf
fe ":WF$
 4600 LET WF$=UCASE$(WF$)
 4605 IF WF$="MESS" AND ORTOBJEKT(3
)=-1 OR ORTOBJEKT(3)=SPIELERRAUM TH
EN
 4610   LET BEMERKUNG=10
 4615   GOTO 4895
 4620 END IF
 4625 !
 4630 IF WF$="MESSER" THEN
 4635   PRINT "Du hast keins!"
 4640   GOTO 1255
 4645 END IF
 4650 !
 4655 IF WF$<>"SPEER" THEN GOTO 478
5
 4660 !
 4665 IF HS=31 AND SPIELERRAUM=4 AN
D ORTOBJEKT(31)=99 THEN
 4670   PRINT "Der Drache ist doch
schon tot!"
 4675   GOTO 1255
 4680 END IF
 4685 !
 4690 IF HS=31 AND SPIELERRAUM=4 AN
D(ORTOBJEKT(6)=SPIELERRAUM OR ORTOB
JEKT(6)=-1) AND ORTOBJEKT(31)<>99 T
HEN PRINT "Der Drache stirbt."
 4695 !
 4700 IF HS=31 AND SPIELERRAUM=4 AN
D(ORTOBJEKT(6)=SPIELERRAUM OR ORTOB
JEKT(6)=-1) AND ORTOBJEKT(31)<>99 T
HEN
 4705   LET ORTOBJEKT(31)=99
 4710   LET DG(4,2)=5
 4715 END IF
 4720 !
 4725 IF HS=31 AND SPIELERRAUM=4 AN
D(ORTOBJEKT(6)=SPIELERRAUM OR ORTOB
JEKT(6)=-1) AND ORTOBJEKT(31)=99 TH
EN GOTO 985
 4730 !
 4735 IF HS=24 THEN LET BEMERKUNG=1
1:GOTO 2150
 4740 !
 4745 IF HS=26 THEN
 4750   PRINT "Eine Giftotter beißt
 dich."
 4755   LET ST=ST-4
 4760   GOTO 1255
 4765 END IF
 4770 !
 4775 PRINT "Der Bär hat ein dickes
 Fell...":GOTO 1255
 4780 !
 4785 PRINT "Das ist keine sehr gut
e Waffe... dein Gegner verletzt dic
h."
 4790 LET STAERKEPUNKTE=STAERKEPUNK
TE-4
 4795 GOTO 1255
 4800 !
```

```
 4805 IF HS=32 AND SPIELERRAUM=13 T
HEN PRINT "Sie murmelt einen ZAUBER
SPRUCH... um"
 4810 IF HS=32 AND SPIELERRAUM=13 T
HEN
 4815   PRINT " dich dreht sich all
es."
 4820   LET SPIELERRAUM=21
 4825   LET DG(21,5)=0
 4830   FOR X=1 TO 100
 4835   NEXT X
 4840 END IF
 4845 !
 4850 IF HS=32 AND SPIELERRAUM=13 T
HEN GOTO 985
 4855 IF HS=26 AND SPIELERRAUM=25 T
HEN PRINT "Den Schlangen wird schle
cht, sie lassen dich durch."
 4860 IF HS=26 AND SPIELERRAUM=25 T
HEN
 4865   LET DG(25,2)=26
 4870   GOTO 985
 4875 END IF
 4880 !
 4885 PRINT "IGITT!":GOTO 1255
 4890 !
 4895 PRINT
 4900 IF BEMERKUNG=1 THEN PRINT "De
in Besuch gefiel dem Drachen nicht.
.."
 4905 IF BEMERKUNG=2 THEN PRINT "Du
 verwandelst dich in eine steinerne
 KRÖTE."
 4910 IF BEMERKUNG=3 THEN PRINT "Di
e Wachen haben dich festgenommen -
DU SITZT IM KERKER."
 4915 IF BEMERKUNG=4 THEN PRINT "DE
INE KRAFT LÄSST NACH... erschöpft b
richst du zusammen."
 4920 IF BEMERKUNG=5 THEN PRINT "Di
e Hunde werden dich hier nicht hera
uslassen."
 4925 IF BEMERKUNG=6 THEN PRINT "Er
 wollte sich nicht nehmen lassen."
 4930 IF BEMERKUNG=7 THEN PRINT "Se
hr tragisch, Dein Selbstmord!!!"
 4935 IF BEMERKUNG=8 THEN PRINT "GR
OOOOOOOOAAAAAAAAR !!!!!"
 4940 IF BEMERKUNG=9 THEN PRINT "Sc
hmeckt gut, aber es sind G I F T -
 KRÄUTER!!!"
 4945 IF BEMERKUNG=12 THEN PRINT "E
in Stein ist dir auf den Kopf gefal
len"
 4950 IF BEMERKUNG=10 THEN PRINT "D
as Messer war zu kurz..."
 4955 IF BEMERKUNG=11 THEN PRINT "D
u hast den 'Roten Eric' nur ein wen
ig gekitzelt"
 4960 IF BEMERKUNG=12 THEN PRINT "E
in Stein ist dir auf den Kopf gefal
len"
 4965 IF BEMERKUNG=13 THEN PRINT "D
er Drache sagt : GIBT NIX !"
 4970 !
 4975 PRINT
 4980 PRINT "=== DEINE MISSION IST
GESCHEITERT ==="
 4985 PRINT "         Versuch es no
chmal !!!"
 4990 PRINT
 4995 PRINT "             ENTER-TAST
E ...";
 5000 INPUT DUMMY$
 5005 GOTO 5475: ! SPIELENDE
 5010 !
 5015 PRINT "  BITTE DISKETTE EINLE
GEN ..."
 5020 PRINT "  DANN ENTER-TASTE.";
 5025 INPUT DUMMY$
 5030 !
 5035 INPUT PROMPT "DATEINAME: ":DD
$
 5040 LET DD$=UCASE$(DD$)
 5045 OPEN #120:DD$ ACCESS OUTPUT
 5050 PRINT #120:SPIELERRAUM
 5055 PRINT #120:STAERKEPUNKTE
 5060 PRINT #120:BONUS
 5065 PRINT #120:ANZBEWEGUNG
 5070 FOR Q=1 TO ANZOBJEKTE
 5075   PRINT #120:ORTOBJEKT(Q)
 5080 NEXT Q
 5085 !
 5090 FOR Q=1 TO ANZFLAGS
 5095   PRINT #120:FLAG(Q)
 5100 NEXT Q
 5105 !
 5110 PRINT #120:DG(4,2)
 5115 PRINT #120:DG(8,4)
 5120 PRINT #120:DG(18,3)
 5125 PRINT #120:DG(18,4)
 5130 PRINT #120:DG(11,5)
 5135 PRINT #120:DG(12,6)
 5140 PRINT #120:DG(21,5)
 5145 PRINT #120:DG(23,1)
 5150 PRINT #120:DG(25,2)
 5155 PRINT #120:DG(22,3)
 5160 CLOSE #120
 5165 !
 5170 !
 5175 PRINT "FERTIG !!":PRINT "SPIE
LENDE (1) WEITER (2)"
 5180 DO
 5185   LET X$=INKEY$
```

```
 5190 LOOP UNTIL X$<>""
 5195 IF X$<"1" OR X$>"2" THEN GOTO
 5180
 5200 IF X$="1" THEN GOTO 5475: ! S
PIELENDE
 5205 IF X$="2" THEN GOTO 985
 5210 GOTO 5180
 5215 !
 5220 PRINT "   BITTE DISKETTE EINL
EGEN ..."
 5225 PRINT "   DANN ENTER-TASTE.";

 5230 INPUT DUMMY$
 5235 !
 5240 INPUT PROMPT "DATEINAME: ":DD
$
 5245 LET DD$=UCASE$(DD$)
 5250 OPEN #120:DD$ ACCESS INPUT
 5255 INPUT #120:SPIELERRAUM
 5265 INPUT #120:STAERKEPUNKTE
 5275 INPUT #120:BONUS
 5285 INPUT #120:ANZBEWEGUNG
 5295 FOR Q=1 TO ANZOBJEKTE
 5300   INPUT #120:ORTOBJEKT(Q)
 5305 NEXT Q
 5310 !
 5315 FOR Q=1 TO ANZFLAGS
 5320   INPUT #120:FLAG(Q)
 5330 NEXT Q
 5335 !
 5340 INPUT #120:DG(4,2)
 5345 INPUT #120:DG(8,4)
 5350 INPUT #120:DG(18,3)
 5355 INPUT #120:DG(18,4)
 5360 INPUT #120:DG(11,5)
 5365 INPUT #120:DG(12,6)
 5370 INPUT #120:DG(21,5)
 5375 INPUT #120:DG(23,1)
 5380 INPUT #120:DG(25,2)
 5385 INPUT #120:DG(22,3)
 5390 CLOSE #120
 5395 !
 5400 PRINT "FERTIG !!":PRINT "SPIE
LENDE (1) WEITER (2)"
 5405 DO
 5410   LET X$=INKEY$
 5415 LOOP UNTIL X$<>""
 5420 IF X$<"1" OR X$>"2" THEN GOTO
 5405
 5425 IF X$="1" THEN GOTO 5475: ! S
PIELENDE
 5430 IF X$="2" THEN GOTO 985
 5435 GOTO 5405
 5440 !
 5445 PRINT "B R A V O   !!!!! DU H
AST GEWONNEN !!!"
 5450 PRINT :PRINT "DU HAST DEN 'RO
TEN ERIC' BESIEGT UND DIE PRINZESSI
N GERETTET !"
 5455 PRINT "AUSSERDEM HAST DU ";BO
NUS-5*ANZBEWEGUNG;" PUNKTE ERZIELT
!"
 5460 GOTO 5475: ! SPIELENDE
 5465 !
 5470 !
 5475 ! **************
 5480 ! * SPIELENDE *
 5485 ! **************
 5490 !
 5495 TEXT 80
 5500 !
 5505 END
```

Enterprise-Reversi

Programmlänge: 8500 Byte
Sprache: IS-BASIC
Zubehör: Enterprise 128

'Reversi' ist ein altes Strategiespiel, welches im 19.Jahrhundert in England erfunden wurde und auch unter dem Namen 'Othello' bekannt ist. Über all' die Jahre hat es aber nichts von seinem Reiz verloren. Vor allem, weil diese Spielversion durch seine intelligenten Bewertungsalgorithmen auch gestandenen Experten den Schweiß auf die Stirn treibt.

Gespielt wird auf einem Schachbrett mit 8*8 Feldern. Als Spielfiguren werden Steine verwendet, die auf der Ober- und Unterseite verschiedene Farben (Schwarz/Weiß) haben. Begonnen wird das Spiel mit einer bestimmten Anfangsaufstellung, die natürlich individuell angepaßt werden kann. Die Zugeingabe erfolgt durch ein Koordinatensystem, wobei sich die zweistellige Zielfeldnummer aus der Summe des vertikalen und des horizontalen Koordinatenwertes ergibt.
Abwechselnd legen dann die Spieler

jeweils einen Stein auf das Brett. Kommen dabei gegnerische Steine zwischen zwei eigene, so werden sie umgedreht und gehören jetzt der eigenen Partei.
Sollte ein Spieler einmal nicht ziehen können, weil er keinen gegnerischen Stein einschliessen kann, so muß er einmal aussetzen (Eingabe von '+'). Abgebrochen wird das Spiel mit '#'.
Gespielt wird solange, bis alle 64 Felder besetzt sind. Gewonnen hat derjenige mit den meisten Steinen in seiner Farbe.
Abschließend noch einige Tips: Wichtig ist, daß man seine Steine in eine der vier unangreifbaren Eckpositionen bekommt. Hier können sie nämlich nicht mehr vom Gegner umgedreht werden und ermöglichen so die Kontrolle der gesamten Spielfeld-Diagonalen.
Ebenfalls interessant sind die vier vorletzten Randreihen, welche nach Umdrehen durch den Gegner eine leichte Rückeroberung erlauben.
Die Spielstrategie des Programms folgt der Methode 'rohe Gewalt'. Der Rechner überprüft also die Feldpositionen auf maximalen Punktgewinn. Die vorletzten Randreihen werden in ihrer Wertigkeit halbiert. Ist etwa ein Zug in die Nähe eines Eckfeldes vorgesehen, wird diese Zahl sogar noch einmal halbiert.
Entsprechend umgekehrt verhält sich die Punktwertung strategisch wertvoller Felder. Randreihen sind doppelt viel Wert und Eckfelder erzielen gar vierfache Wertungen.

Reiner Kunz

```
100 PROGRAM "Reversi"
110 !
120 NUMERIC FELD(10,10)
130 NUMERIC DZ(10)
140 NUMERIC DSP(10)
150 LET X=0
160 LET PR=0
170 TEXT 40
180 CALL ARRAYINIT
190 CALL CREATE_CHARSET
200 CALL SPIELREGELN
210 GOTO 1470 ! Initialisierung
220 !
230 ! ************************
240 ! * der Computer spielt *
250 ! ************************
260 !
270 PRINT AT 19,50:"Mein Zug ...
          "
280 LET S=0
290 LET T=X
300 LET M=0
310 FOR A=2 TO 9
320   FOR B=2 TO 9
330     IF FELD(A,B)<>PKT THEN GO
TO 570
340     LET Q=0
350     FOR C=-1 TO 1
360       FOR D=-1 TO 1
370         LET K=0
380         LET F=A
390         LET G=B
400         IF FELD(F+C,G+D)<>S T
HEN GOTO 450
410         LET K=K+1
420         LET F=F+C
430         LET G=G+D
440         GOTO 400
450         IF FELD(F+C,G+D)<>T T
HEN GOTO 470
460         LET Q=Q+K
470       NEXT D
480     NEXT C
490     !
500     IF A=2 OR A=9 OR B=2 OR B
=9 THEN LET Q=Q*2
510     IF (A=2 OR A=9) AND(B=3 O
R B=8) OR(A=3 OR A=8) AND(B=2 OR B=
9) THEN LET Q=Q/2
520     IF (A=2 AND B=2) OR(A=2 A
ND B=9) OR(A=9 AND B=2) OR(A=9 AND
B=9) THEN LET Q=Q*2
530     IF Q<M OR(RND(1)<.3 AND Q
=M) THEN GOTO 570
540     LET M=Q
550     LET P=A
560     LET N=B
570   NEXT B
580 NEXT A
590 !
600 IF M=0 THEN GOTO 630
610 !
620 CALL UMDREHEN
630 CALL SPIELSITUATION
640 !
650 ! ************************
660 ! * Spiel des Spielers *
670 ! ************************
680 !
690 PING
700 PRINT AT 19,50:;:CALL STRING(
20,CHR$(32))
710 PRINT AT 19,50:"Ihr Zug, bitt
e : ";
720 DO
730   LET A$=INKEY$
740 LOOP UNTIL A$<>""
750 IF A$="+" THEN GOTO 1020
760 IF A$="#" THEN GOTO 3120
770 LET R=VAL(A$)
780 IF R<1 OR R>8 THEN
790   GOTO 720
800 ELSE
810   PRINT A$;
820 END IF
830 DO
840   LET A$=INKEY$
850 LOOP UNTIL A$<>""
860 LET Z=VAL(A$)
870 IF Z<1 OR Z>8 THEN
880   GOTO 830
890 ELSE
900   PRINT A$
910 END IF
920 LET R=10*R+Z
930 LET R=R+11
940 LET S=X
950 LET T=0
960 CALL PRUEFE_REGEL
970 IF PR=0 THEN GOTO 690
980 !
990 LET P=INT(R/10)
1000 LET N=R-10*P
1010 CALL UMDREHEN
1020 CALL SPIELSITUATION
1030 GOTO 230
1040 !
1050 ! ****************************
1060 ! * Spielsituation ausgeben *
1070 ! ****************************
1080 !
1090 DEF SPIELSITUATION
1100   LET C=0
1110   LET M=0
1120   FOR B=2 TO 9
1130     FOR D=2 TO 9
1140       PRINT AT 2*(B-1)+1,4*(D
-1)+2:CHR$(FELD(B,D));
1150       IF FELD(B,D)=X THEN LET
C=C+1
1160       IF FELD(B,D)=O THEN LET
M=M+1
1170     NEXT D
1180   NEXT B
1190   PRINT AT 21,50:"Computer ";
CHR$(128);":";C
1200   PRINT AT 22,50:"Spieler  ";
CHR$(129);":";M
1210   GOSUB 2980 !Prüfen auf Ende
1220 END DEF
1230 !
1240 ! ************************
1250 ! * Spielsteine umdrehen *
1260 ! ************************
1270 !
1280 DEF UMDREHEN
1290   FOR C=-1 TO 1
1300     FOR D=-1 TO 1
1310       LET F=P
1320       LET G=N
1330       IF FELD(F+C,G+D)<>S THE
N GOTO 1370
1340       LET F=F+C
1350       LET G=G+D
1360       GOTO 1330
1370       IF FELD(F+C,G+D)<>T THE
N GOTO 1430
```

```
 1380       LET FELD(F,G)=T
 1390       IF P=F AND N=G THEN GOT
O 1430
 1400       LET F=F-C
 1410       LET G=G-D
 1420       GOTO 1380
 1430     NEXT D
 1440   NEXT C
 1450 END DEF
 1460 !
 1470 ! ********************
 1480 ! * Initialisierung *
 1490 ! ********************
 1500 !
 1510 LET PKT=32
 1520 LET X=ORD(CHR$(128)) ! Spielm
arke für Computer
 1530 LET O=ORD(CHR$(129)) ! Spielm
arke für Mensch
 1540 !
 1550 FOR B=1 TO 10
 1560   FOR C=1 TO 10
 1570     IF B<>1 AND C<>1 AND B<>1
0 AND C<>10 THEN LET FELD(B,C)=PKT
 1580   NEXT C
 1590 NEXT B
 1600 !
 1610 LET FELD(5,5)=X
 1620 LET FELD(6,6)=X
 1630 LET FELD(6,5)=O
 1640 LET FELD(5,6)=O
 1650 !
 1660 LET L=1
 1670 FOR C=-1 TO 1
 1680   FOR D=-1 TO 1
 1690     LET DZ(L)=C
 1700     LET DSP(L)=D
 1710     LET L=L+1
 1720   NEXT D
 1730 NEXT C
 1740 !
 1750 LET R=0
 1760 PRINT
 1770 PRINT "Wollen Sie beginnen'?
(J/N) ";
 1780 LET W$=UCASE$(INKEY$)
 1790 IF W$="" THEN GOTO 1780
 1800 TEXT 80
 1810 CALL SPIELFELD
 1820 CALL SPIELSITUATION
 1830 IF W$="J" THEN GOTO 650
 1840 GOTO 270
 1850 !
 1860 ! **************
 1870 ! * Spielfeld *
 1880 ! **************
 1890 !
 1900 DEF SPIELFELD
 1910   NUMERIC I,J
 1920   CLEAR SCREEN
 1930   !
 1940   PRINT "   ";
 1950   FOR I=1 TO 8
 1960     PRINT CHR$(130);I;
 1970   NEXT I
 1980   PRINT CHR$(130)
 1990   !
 2000   FOR J=1 TO 8
 2010     FOR I=1 TO 9
 2020       CALL STRING(3,CHR$(131)
)
 2030       PRINT CHR$(132);
 2040     NEXT I
 2050     !
 2060     CALL STRING(3,CHR$(131))
 2070     PRINT
 2080     !
 2090     FOR I=1 TO 9
 2100       PRINT "   ";CHR$(130);
 2110     NEXT I
 2120     PRINT
 2130   NEXT J
 2140   !
 2150   FOR I=1 TO 9
 2160     CALL STRING(3,CHR$(131))
 2170     PRINT CHR$(132);
 2180   NEXT I
 2190   CALL STRING(3,CHR$(131))
 2200   PRINT
 2210   PRINT "   ";
 2220   FOR I=1 TO 8
 2230     PRINT CHR$(130);I;
 2240   NEXT I
 2250   PRINT CHR$(130)
 2260   !
 2270   FOR J=10 TO 80 STEP 10
 2280     PRINT AT 2*J/10+1,1:J
 2290     PRINT AT 2*J/10+1,37:J
 2300   NEXT J
 2310   !
 2320   FOR J=1 TO 8
 2330     PRINT AT 2*J+1,4:CHR$(130
)
 2340   NEXT J
 2350 END DEF
 2360 !
 2370 !
 2380 ! ****************************
**
 2390 ! * Erklärung der Spielregeln
 *
 2400 ! ****************************
**
 2410 !
 2420 DEF SPIELREGELN
 2430   CLEAR SCREEN
 2440   PRINT "          R E V E R
S I"
 2450   PRINT "          ==========
==="
 2460   PRINT
 2470   PRINT "Spieler und Rechner
setzen abwechselnd"
 2480   PRINT "einen Stein auf ein
Schachbrett."
 2490   PRINT "Dies muß so erfolgen
, daß mindestens"
 2500   PRINT "ein Stein des Gegner
eingeschlossen"
 2510   PRINT "werden kann. Dieses
kann horizontal,"
 2520   PRINT "vertikal oder diagon
al erfolgen."
 2530   PRINT "Die jeweils eingesch
lossenen Steine"
 2540   PRINT "werden umgedreht und
damit also"
 2550   PRINT "zu eigenen Steinen."

 2560   PRINT "Kann ein Spieler nic
ht ziehen, so ist"
 2570   PRINT "der andere erneut an
der Reihe."
 2580   PRINT "Der Spieler muß hier
zu ein '+' "
 2590   PRINT "eingeben."
 2600   PRINT "Wollen Sie vorzeitig
aufhören, so"
 2610   PRINT "geben Sie bitte '#'
ein"
 2620   PRINT
 2630   PRINT "        V I E L  S P
A S S"
 2640   PRINT "        ============
======"
 2650   !
 2660 END DEF
 2670 !
 2680 ! ****************************

 2690 ! * prüfe, ob Regel gewahrt *

 2700 ! ****************************

 2710 !
 2720 DEF PRUEFE_REGEL
 2730   LET ZL=INT(R/10)
 2740   LET SP=R-(ZL*10)
 2750   LET GE=0
 2760   IF FELD(ZL,SP)<>PKT THEN LE
T PR=0:GOTO 2960
 2770   !
 2780   FOR AA=1 TO 9
 2790     LET FF=1
 2800     LET EE=0
 2810     IF NOT(FELD(ZL+FF*DZ(AA),
SP+FF*DSP(AA))=S) THEN
 2820       GOTO 2890
 2830     ELSE
 2840       LET FF=FF+1
 2850       LET EE=EE+1
 2860       GOTO 2810
 2870     END IF
 2880     !
 2890     IF FELD(ZL+FF*DZ(AA),SP+F
F*DSP(AA))=T THEN LET GE=GE+EE
 2900   NEXT AA
 2910   IF GE>0 THEN
 2920     LET PR=1
 2930   ELSE
 2940     LET PR=0
 2950   END IF
 2960 END DEF
 2970 !
 2980 ! ******************
 2990 ! * Prüfe auf Ende *
 3000 ! ******************
 3010 !
 3020 LET PP=0
 3030 FOR ZZ=2 TO 9
 3040   FOR SS=2 TO 9
 3050     IF FELD(ZZ,SS)<>PKT THEN
LET PP=PP+1
 3060   NEXT SS
 3070 NEXT ZZ
 3080 IF PP<64 THEN RETURN
 3090 IF C>M THEN PRINT AT 22,5:"Si
e haben leider verloren!"
 3100 IF M>C THEN PRINT AT 22,5:"Br
avo, Sie haben gewonnen!"
 3110 IF M=C THEN PRINT AT 22,5:"Un
entschieden!"
 3120 IF A$="#" THEN PRINT AT 22,5:
"Schade, daß Sie aufhören wollen."
 3130 PRINT AT 21,50:;:CALL STRING(
25,CHR$(32))
 3140 !
 3150 ! **********************
 3160 ! * Ende des Programms *
 3170 ! **********************
 3180 !
 3190 CLEAR FONT
 3200 EXT "BRD"
 3210 TEXT 80
 3220 END
 3230 !
 3240 !
 3250 ! ****************************
**********************
 3260 ! * Mache beliebigen String a
us beliebigen Zeichen *
 3270 ! ****************************
**********************
 3280 !
 3290 DEF STRING(LENGTH,CHAR$)
 3300   NUMERIC ZAEHLER
 3310   !
 3320   FOR ZAEHLER=1 TO LENGTH
 3330     PRINT CHAR$;
 3340   NEXT ZAEHLER
 3350   !
 3360 END DEF
 3370 !
 3380 ! ****************************
*****
 3390 ! * Deklariere neuen Zeichens
atz *
 3400 ! ****************************
*****
 3410 !
 3420 DEF CREATE_CHARSET
 3430   RESTORE 3590
 3440   NUMERIC N(9)
 3450   NUMERIC NUMBER
 3460   NUMERIC ZEILE
 3470   !
 3480   FOR NUMBER=128 TO 132
 3490     FOR ZEILE=1 TO 9
```

```
 3500       READ N(ZEILE)
 3510     NEXT ZEILE
 3520     !
 3530     SET CHARACTER NUMBER,BIN(
N(1)),BIN(N(2)),BIN(N(3)),BIN(N(4))
,BIN(N(5)),BIN(N(6)),BIN(N(7)),BIN(
N(8)),BIN(N(9))
 3540     !
 3550   NEXT NUMBER
 3560   !
 3570   ! Spielmarke Computer / #12
8
 3580   !
 3590   DATA 00000000
 3600   DATA 00011100
 3610   DATA 00111110
 3620   DATA 01111111
 3630   DATA 01111111
 3640   DATA 01111111
 3650   DATA 00111110
 3660   DATA 00011100
 3670   DATA 00000000
 3680   !
 3690   ! Spielmarke Mensch / #129
 3700   !
 3710   DATA 00000000
 3720   DATA 00011100
 3730   DATA 00111110
 3740   DATA 01100011
 3750   DATA 01100011
 3760   DATA 01100011
 3770   DATA 00111110
 3780   DATA 00011100
 3790   DATA 00000000
 3800   !
 3810   ! Spielfeldzeichen / #130
 3820   !
 3830   DATA 00011000
 3840   DATA 00011000
 3850   DATA 00011000
 3860   DATA 00011000
 3870   DATA 00011000
 3880   DATA 00011000
 3890   DATA 00011000
 3900   DATA 00011000
 3910   DATA 00011000
 3920   !
 3930   ! Spielfeldzeichen / #131
 3940   !
 3950   DATA 00000000
 3960   DATA 00000000
 3970   DATA 00000000
 3980   DATA 11111111
 3990   DATA 11111111
 4000   DATA 11111111
 4010   DATA 00000000
 4020   DATA 00000000
 4030   DATA 00000000
 4040   !
 4050   ! Spielfeldzeichen / #132
 4060   !
 4070   DATA 00011000
 4080   DATA 00011000
 4090   DATA 00011000
 4100   DATA 11111111
 4110   DATA 11111111
 4120   DATA 11111111
 4130   DATA 00011000
 4140   DATA 00011000
 4150   DATA 00011000
 4160 END DEF
 4170 !
 4180 ! ****************************
**
 4190 ! * Array Feld initialisieren
 *
 4200 ! ****************************
**
 4210 !
 4220 DEF ARRAYINIT
 4230   NUMERIC I,J
 4240   !
 4250   FOR I=0 TO 10
 4260     FOR J=0 TO 10
 4270       LET FELD(I,J)=0
 4280     NEXT J
 4290   NEXT I
 4300   !
 4310 END DEF
```

Buchtips

Es ist selbstverständlich, daß wir Ihnen zum Schluß unseres Specials noch ein paar Buchtips geben. Der Bücherberg zum Thema BASIC ist schon fast unüberschaubar groß geworden. Fast jeder Computer ist vertreten. Wenn man aber das Bücherangebot etwas genauer unter die Lupe nimmt, finden sich kaum allgemeingültige Titel. Oberflächliche Behandlungen der Programmiersprache BASIC gibt es in rauhen Mengen.
Der Enterprise-Fan ist auch in anderer Hinsicht gehandicapt: Sein Rechner ist so brandneu, daß vielen Verlagen keine Zeit blieb, spezielle Literatur zu veröffentlichen.
Darum sollen an dieser Stelle einige gute und allgemeinverständliche Titel besprochen werden, die auch nach vielen Monaten noch nicht wertlos im Regal verstauben.

Datenstrukturen in Pascal und BASIC

von Dietmar Herrmann
Vieweg 1984; 58 S.; kart.;DM 19.80
ISBN 3 528 04263 X

Wegen der starken Betonung von algorithmischen Verfahren in Literatur und Ausbildung ist die Darstellung von Datenstrukturen ein wenig in den Hintergrund getreten. Doch gerade die Wahl der richtigen Datentypen ist oft mitentscheidend über die Effizienz von Algorithmen. Ältere Versionen von BASIC besaßen nur wenig Auswahl an verschiedenen Datentypen. Aber gerade das Enterprise-BASIC macht da eine begrüßenswerte Ausnahme. Durch den Vergleich mit Pascal lernt der Leser viel nützliches über Aufzählungs- und Unterbereichstypen, Mengen, Verbunde, Felder, Listen, Stacks (Stapelspeicher), Schlangen, Bäume und Heaps.
Alle diese verschiedenen Datentypen werden anhand von 12 Pascal- und 8 BASIC-Programmen erläutert.
Für jemanden, der sich ein wenig tiefer in die Materie 'Datenverarbeitung' hineinarbeiten will, ein empfehlenswertes Buch.

Programmierprinzipien in Basic

von Dietmar Herrmann
Vieweg 1984; 59 S.; kart.;DM 19.80
ISBN 3 528 04258 3

Das Entwerfen von Algorithmen für die Programmierung von Computern wird wohl auch einem Anfänger (bei einfachen Aufgaben) nicht schwerfallen. Doch wenn es um Effizienz und komplexere Aufgabenstellungen geht, stößt man schnell an die Grenzen seines Könnens.
Hilfe bietet dieses Buch, welches etwa ein Dutzend grundlegender Programmiertechniken wie Top-Down-Entwurf, Rekursion, Iteration, Backtracking-Verfahren und Greedy-Algorithmen vorstellt.
Anhand von 12 Pascal- und 12 BASIC-Programmen werden ihre Funktionsweisen erläutert. Neben numerischen und mathematischen Problemen kommen auch kombinatorische und Sortier-Fragen zum Zuge.

BASIC-Programmierbuch

von Ekkehard Kaier
Vieweg 1983; 185 S.; kart.; DM 32
ISBN 3 528 04222 2

Dieses Buch wendet sich an alle, die mehr vom Programmieren wissen wollen, als das stupide Abtippen ellenlanger Listings. Um dem Leser die Vorgehensweise bei der Erstellung eigener Programme zu demonstrieren, werden 46 komplette Programmbeispiele systematisch zerpflückt und ausführlich erläutert.
Der Problemstellung folgen die Problemanalyse, die zeichnerische Darstellung mit Programmablaufplan und Struktogramm, die Überführung in den BASIC-Quelltext, die Programmausführung mit Computer- und Schreibtischtest und anschließenden Fragen, die das Verständnis des Programms vertiefen helfen.
Außerdem werden folgende Themen behandelt:
binäres Suchen, Sortierverfahren, Mischverfahren, Abschreibungen, Zinsrechnung, Terminrechnen usw.

BASIC-Trainer

von Hansrobert Kohler
Vieweg 1984; 140 S.; kart.; DM 28
ISBN 3 528 04306 7

Dieses stark naturwissenschaftlich-mathematisch orientierte BASIC-Buch ist eine ausgesprochene Übungssammlung mit kompletten Lösungsbeschreibungen. Aus allen Gebieten der Naturwissenschaft finden sich interessante Aufgabenstellungen.
Darum ist dieses Buch auch eher für diejenigen BASIC-Programmierer gedacht, die bereits gute Grundkenntnisse besitzen und ihr Wissen an praxisnahen Übungen prüfen wollen.

BASIC

von Max Riederle und Siegfried Stief
Oldenbourg 1984; 330 S.; Paperback
ISBN 3 486 28681 1

In den letzten Jahren gab es starke Bestrebungen, den BASIC-Wildwuchs (jeder Hersteller baut seinen eigenen Dialekt) einzudämmen. Das Ergebnis liegt jetzt vor: ANSI-BASIC. Dieses Buch ist die erste ausführliche Darstellung des amerikanischen Norm-Entwurfs in der Fassung vom 20.6.1980. Durch die systematische Gliederung kann es lehrbuchmäßig von vorne bis hinten durchgearbeitet werden.
Leider ist es stellenweise etwas konfus und oberflächlich geschrieben, so daß ein zügiges Lesen teilweise nicht möglich ist.
Wer sich jedoch etwas tiefer in die Ursprünge des IS-BASIC von Enterprise einarbeiten möchte, dem sei dieser Titel ans Herz gelegt. Da der ANSI-Norm-Entwurf sich aber doch zum Teil sehr stark vom IS-BASIC unterscheidet ist dieses Werk eine gute Studie darüber, was Hersteller von Norm-Entwürfen halten: nämlich so gut wie nichts. Darum kann diese Buch auch nur mit gewissen Einschränkungen empfohlen werden.

Programmierung des Z80

von Rodnay Zaks
Sybex-Verlag 1983; 606 S.; Paperback
ISBN 3-88745-006-X

Dieses Buch über die Programmierung des Z80 in Maschinensprache gehört zu den Klassikern der Mikrocomputer-Literatur.
Es führt sehr geschickt in die Grundprinzipien der Assemblersprache ein und läßt nie den bezug zur Hardware außer acht. Die Theorie (binäre Arithmetik, Suchalgorithmen usw.) wird ebenso ausführlich behandelt, wie praktische Programmierarbeit. Für Einsteiger und Fortgeschrittene auf dem Gebiet der Maschinenprogrammierung ist dieses Buch ein absolutes Muß.

Vom Umgang mit CP/M

von Bernd Pol
IWT-Verlag 1985; 376 S., Hardcover
ISBN 3-88 322-004-3

Sicherlich haben Sie schon gehört, daß der Enterprise aufgrund seines Z80-Prozessors CP/M-tauglich ist. Die entsprechende Systemsoftware heißt IS-DOS und wird beim Kauf einer Diskettenstation auf Wunsch zugesandt.
Da man bei der Arbeit unter CP/M sehr schnell an die Grenzen des Informationsgehalts der mitgelieferten Handbücher stößt, sollte man sich, um sein Wissen um dieses universelle Betriebssystem zu erweitern, das vorliegende Buch besorgen. Mit penibler Genauigkeit wird der Leser vom Autor in die Geheimnisse diese Systems eingeführt ohne dabei langatmig zu werden. Störend ist allein, daß wirklich Fortgeschrittene nur wenig Hinweise und Tips für ihr Hobby aus diesem Buch entnehmen können. Auch ist es unverständlich, warum ganze Kapitel mit der Beschreibung unbedeutender oder gar veralteter Systemutilities (z.B. ED, DDT und ASM für 8080) verschwendet werden. Besser wären Beschreibungen für Z80-orientierte Programme (z.B. DDTZ, M80, L80). Trotz dieser Mängel, bleibt dieses Buch bis jetzt das einzig brauchbare für CP/M auf dem deutschen Markt.

Grundzüge der interaktiven Computergraphik

von Newmann/Sproull

McGraw-Hill 1986; 582 S.; Paperback
ISBN 3-89028-015-3

Die Bedeutung der Computergrafik in Naturwissenschaft/Technik sowie im modernen Wirtschaftsleben gewinnt stetig an Bedeutung. Wer hier mitreden will oder muß, dem sei diese Übersetzung des US-Bestsellers wärmstens empfohlen. Doch auch, wer die Grafikfähigkeiten seines Enterprise voll ausschöpfen möchte, sollte sich mit diesem Buch beschäftigen. Besprochen werden folgende Themen unter Angabe der dazugehörigen Algorithmen:
Punktdarstellungstechniken, Bildschirmtechnik, zwei- und drei-dimensionale Bildtransformationen, Clipping, Windowing, Bildstrukturen, Rastergrafik, Unterdrückung verdeckter Linien, Schattieren usw.

Reiner Kunz

Händler in Ihrer Nähe

Aval Produkt & Vertriebs GmbH
Kantstraße 148
1000 Berlin 12
030/310462

Computing
Im U-Bhf. Schloßstraße
1000 Berlin 41
030/7920220

Elektroschach-Heide Ketterling
Dudenstraße 32
1000 Berlin 61
030/7857674

Ihre Computerei
Karl-Marx-Straße 243
1000 Berlin 44
030/6841098

KMB Unternehmensberatung
Hermannstraße 9
1000 Berlin 44
030/6212067

Mötz, Wolfgang
Friedrichstraße 235
1000 Berlin 61
030/2516001

Schöpplenberg, Marc
Potsdammer Straße 63
1000 Berlin 30
030/2626271

Brückner Elektronik
Schulstraße 1
2111 Königsmoor
04180/236

Reese GmbH & Co. KG
Flensburger Landstraße 196
2300 Kiel 1
0431/6891276

Hamelberg, Fritz
Bartelsdorfer Straße 1
2720 Rothenburg/Wümme
04261/67712

Datalogik
Calenbergstraße 26
3000 Hannover 1
0511/326489

Lendorf Computer EDV
Goldener Winkel 18
3013 Barsinghausen 4
05105/64391

Brauer Hard- und Software
Neustadtstraße 25
3240 Kappeln
04642/81893

Dinkelbach Computer
Schöffenstedter Straße 26
3300 Braunschweig
0531/40167

Bi-Ra Vertrieb
Obere Straße 50–52
3450 Holzminden
05531/61724

Schilling Elektronik
Kölner Landstraße 192
4000 Düsseldorf 13
0211/762100

Schmitz, Thomas EDV
Vitusstraße 25
4050 Möchengladbach 1
02161/206687

Computeragentur Nachtsheim
Haagstraße 4
4130 Moers 1
02841/26963

Softwarebüro Knaack
Rankauerweg 11
4406 Drensteinfurt
02508/397

Kleinertz
Paul-Klee-Weg 16
4630 Bochum 6
02327/73131

Röth Datensysteme
Saarbrücker Straße 40
5600 Wuppertal
0202/5840

HEW Computer Vertriebs GmbH
Wasserbank 9
5810 Witten 3
02302/77353

Compulit
Berliner Straße 66
6000 Frankfurt/Main 1
069/281816

Tischer, Siegfried
Eröffnung Mitte 1986
6000 Frankfurt/Main
069/6708902

SOR Computer Systeme
Lessingstraße 14
6843 Biblis
06245/3325

Comput Elektronik
Im Heppacher 17
7300 Esslingen
0711/354643

Computerstudio Wecker
Hafenbad 18
7900 Ulm
0731/68076

Foto Siegel
Augsburger Straße 100
7910 Neu Ulm
0731/75084

Hobby-Elektronik
Memminger Straße 34
7955 Ochsenhausen
07352/3138

Zaich Softwarebüro
Holdersteige 7
7965 Ostrach
07585/1776

Computec
Planegger Straße 1
8000 München 60
089/838553

Media Markt
Ingolstädter Straße 62
8000 München 45
089/3164026

Mooser Meß- und Datentechnik
Peschelanger 14
8000 München 83
089/263832

Schmid Electronics
Sonnenstraße 3
8000 München 2
089/557170

Media Markt
Erlagstraße 23
8070 Ingolstadt
0841/6340

Media Markt
Georg-Aicher-Straße 21
8200 Rosenheim
08031/43091

Computer Center Deutsch
Jahnstraße 36
8220 Traunstein
0861/4444/3

Media Markt
Gaußstraße 1
8300 Landshut
0871/74214

Computerstudio Passau
Eröffnung März 1986
8390 Passau
08505/2526

Zimmermann Elektroland
Meraner Straße 5
8390 Passau
0851/51061

Zimmermann Elektroland
Dr.-Gessler-Straße 8
8400 Regensburg
0941/95085

CHIP SPECIAL

1. Auflage 1986

Best.-Nr. 0220

Redaktionsdirektor: Richard Kerler

Chefredakteur: Armin Schwarz, verantwortlich für den Inhalt

Redaktionsservice: Paula Rath

Programmservice und Koordination: Franz Joseph Schreiner

Test: Ulrich Kern

Gestaltung: Hans Kuh, Gabi Klotz

Titelillustration: Ernst Jünger

Mitarbeiter dieser Ausgabe: Reiner Kunz, Thomas Müller

Redaktion: Vogel-Verlag KG Würzburg, Redaktion CHIP-Special, Schillerstr. 23a, D-8000 München 2, Telefon (089) 514930, Telekopierer 535000, Telex 5216449

Verlag: Vogel-Verlag KG, Postfach 6740, D-8700 Würzburg 1, Tel. (0931) 4102-1, Telex 68883, Telefax (0931) 4102-529. Telegramme: CHIP-Würzburg

Verlagsdirektor: Dipl.-Kfm. Herbert Frese

Anzeigenleiter: Harald Kempf, Würzburg (verantwortlich für Anzeigen)

Anzeigenservice: CHIP, Postfach 6740, 8700 Würzburg 1, Tel. (0931) 4102-1, Telex 68883, Michael Belgrad, Durchwahl 4102-433. PLZ 1–5 und Ausland: Christine Himmer und Wolfgang Hartmann, Durchwahl 4102-227 PLZ 6–8: Angelika Hirsch und Axel Winheim, Durchwahl 4102-513

Vertriebsleitung: Axel Herbschleb, Würzburg

Vertrieb Handelsauflage: Vereinigte Motor-Verlage GmbH & Co. KG, Leuschnerstr. 1, D-7000 Stuttgart 1, Tel. (0711) 2043-1

Bezugsmöglichkeiten: Bestellungen nehmen der Verlag und alle Buchhandlungen im In- und Ausland entgegen. Sollte die Zeitschrift aus Gründen, die nicht vom Verlag zu vertreten sind, nicht geliefert werden können, besteht kein Anspruch auf Nachlieferung oder Erstattung vorausbezahlter Bezugsgelder.

Bankverbindungen Vogel-Verlag:
Dresdner Bank AG, Würzburg
(BLZ 79080052) 3148890000,
Bay. Vereinsbank AG, Würzburg
(BLZ 79020076) 2506173,
Kreissparkasse Würzburg
(BLZ 79050130) 17400,
Postscheckkonto Nürnberg
(BLZ 76010085) 9991-853
Ausland: Postscheckkonto Zürich 8047064,
Niederlande 2662395
Banque Veuve Morin-Pons, Paris 155410314

Gesamtherstellung und Versand: VOGEL-DRUCK WÜRZBURG, Max-Planck-Str. 7/9, D-8700 Würzburg

Unverlangte Manuskripte werden nur zugesandt, wenn Rückporto beigefügt ist.
Für die mit Namen oder Signatur des Verfassers gekennzeichneten Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung.
Die in dieser Zeitschrift veröffentlichten Beiträge sind urheberrechtlich geschützt. Übersetzung, Nachdruck, Vervielfältigung sowie Speicherung in Datenverarbeitungsanlagen nur mit ausdrücklicher Genehmigung des Verlages. Jede im Bereich eines gewerblichen Unternehmens hergestellte oder benutzte Kopie dient gewerblichen Zwecken gem. § 54 (2) UrhG und verpflichtet zur Gebührenzahlung an die VG Wort, Abteilung Wissenschaft, Goethestraße 49, 8000 München 2, von der die Zahlungsmodalitäten zu erfragen sind.
Die Redaktion hat die Manuskripte und Programme sorgfältig geprüft. Für Fehler im Text, in Schaltbildern, Aufbauskizzen, Listings usw. sowie deren Folgen kann keine Haftung übernommen werden. Sämtliche Veröffentlichungen erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes, auch werden Warennamen ohne Gewährleistung einer freien Verwendung benutzt.

Vervollständigen Sie noch heute Ihre Software-Bibliothek!

Bestellen Sie die Ihnen noch fehlenden CHIP-SPECIALS. Mit den nebenstehenden Karten geht's ganz leicht. Einfach ausfüllen und ab die Post!

1 Bestellkarte für weitere CHIP-SPECIALS

☐ **Ja,** senden Sie mir bitte die angekreuzten SPECIALS zu den genannten Preisen zuzüglich Versandkostenanteil DM 3,50 im Inland. (Versandkostenanteil für das Ausland DM 6,–)
Ich bezahle erst, wenn ich Ihre Rechnung erhalten habe.

Anzahl	Titel	Best.-Nr.	DM/St.
	Atari ST	0230	28,–
	M-Basic	0200	28,–
	Sharp PC 2500, 1500, 1401/2, 1350	0160	28,–
	Sharp MZ 7xx, 8xx	0030	28,–
	Turbo-Pascal, Ausgabe 1	0120	28,–
	IBM PC und Kompatible, Ausgabe 2	0070	28,–
	Epson HX 20	930	28,–
	Commodore Amiga	0290	28,–

Anzahl	Titel	Best.-Nr.	DM/St.
	CHIP-Testjahrbuch/Hardware	0240	24,–
	CHIP-Testjahrbuch/Software	0250	24,–
	Neue Medien	0050	24,–
	Telekommunikation	0100	24,–
	MSX-Computer, Ausgabe 1	0140	24,–
	Die schönsten Elektronik-hobbys	0190	24,–

0690

Datum

Unterschrift

Bitte genaue Anschrift auf der Rückseite angeben.

2 Bestellkarte für weitere CHIP-SPECIALS

nur direkt über den Vogel-Verlag zu bestellen:

☐ **Ja,** senden Sie mir bitte die angekreuzten SPECIALS zu den genannten Preisen zuzüglich Versandkostenanteil DM 3,50 im Inland. (Versandkostenanteil für das Ausland DM 6,–).
Ich bezahle erst, wenn ich Ihre Rechnung erhalten habe.

Anzahl	Titel	Best.-Nr.	DM/St.
	Thomson TO 7-70 und MO 5E	0210	18,–
	Philips „Yes"	0330	28,–
	Computer Programm-Verzeichnis 1986	0320	16,80

0695

Datum

Unterschrift

Bitte genaue Anschrift auf der Rückseite angeben.

CHIP

Bestellen Sie noch heute die Programme für den Enterprise

3 Bestellkarte für CHIP-SOFT Enterprise-Programme

0070

Bitte senden Sie mir alle Programme dieser Ausgabe

☐ auf Kassette für DM 70,–

☐ auf 3,5-Zoll-Diskette für DM 85,–

jeweils inkl. MwSt. plus DM 3,50 Versandkostenanteil und DM 3,20 für Nachnahme – Inland.
(Ausland: DM 6,– Versandkostenanteil plus Nachnahme).

Datum

Unterschrift

Bitte genaue Anschrift auf der Rückseite angeben.

Name, Vorname

Straße/Postfach

PLZ/Ort

Bitte mit 60 Pfennig freimachen

Antwort

Vogel-Verlag
CHIP-SPECIAL
Leser-Service 735
Postfach 6740

D-8700 Würzburg 1

CHIP SPECIAL

Name, Vorname

Straße, Postfach

PLZ/Ort

Ich könnte Ihnen für folgende(n) Computer

folgende selbstentworfene Programme anbieten:

Bitte mit 60 Pfennig freimachen

Antwort

Vogel-Verlag
CHIP-SPECIAL
Leser-Service 735
Postfach 6740

D-8700 Würzburg 1

Name, Vorname

Straße, Postfach

PLZ/Ort

Bitte mit 60 Pfennig freimachen

Antwort

CHIP-SHOP
Leser-Service 735
Vogel-Verlag
Postfach 6740

D-8700 Würzburg 1

Manfred Czerwinski

Mikrocomputer-Pannenhelfer

Hard- und Softwarefehler erkennen · beheben · vermeiden

CHIP WISSEN

Die Freude an Ihrem Computer währt nur so lange, wie er reibungslos funktioniert. Treten jedoch die fast unvermeidlichen hard- oder softwarebedingten Pannen auf, ist guter Rat teuer.

Mit diesem Buch können Sie die Fehler am Computer, Bildschirm und Drucker sowie in Ihren Programmen aufspüren, Abhilfe schaffen und weiteren Fehlfunktionen vorbeugen. Die Bereiche Hardware und Software werden getrennt behandelt. Eine ausführliche Fehlersuchtabelle schließt den Hardware-Teil ab; im Software-Teil werden die Standard-Fehlermeldungen verschiedener BASIC-Versionen erläutert, logische Fehler behandelt und diverse Testmethoden vorgestellt. Ein Lexikon der wichtigsten Fachbegriffe im Anhang rundet die Thematik ab.

VOGEL-BUCHVERLAG
WÜRZBURG

In dieser Ausgabe

Grafik	**Sierpinski-Kurven** **Hilbert-Kurven**
Lernen	**Vokabeltrainer**
Mathematik	**Integralrechnen**
Hobby	**Mein Biorhythmus** **Life is life**
Small Business	**Adressen**
Datenbank	**Universal-Datei**
Statistik	**Polynome**
Steuererklärung	**Abschreibung**
Physik	**Teilchenbahnen**
Biologie	**Räuber-Beute-Population**
Rekursion	**Hanoi-Türme in IS-BASIC**
Denkspiel	**Logic**
Kartenspiel	**Skatrunde**
Adventure	**Der Rote Eric**
Strategie	**Enterprise-Reversi**